EXHIBIT 2

REDACTED

IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF VIRGINIA
ALEXANDRIA DIVISION

UNITED STATES OF AMERICA, et al.,
Plaintiffs,

v.

GOOGLE, LLC,
Defendant.

Case No. 1:23-cv-00108 (LMB/JFA)

HON. LEONIE H. BRINKEMA

EXPERT REPORT OF GORANKA BJEDOV

July 7, 2025

I. ASSIGNMENT

1. I have been retained by the Antitrust Division of the United States Department of Justice ("Counsel") to provide expert testimony in the matter captioned United States of America et al., v. Google, LLC (No. 1:23-cv-00108 (LMB/JFA)) regarding certain technical aspects of Plaintiffs' proposed remedies.

2. Counsel has asked me to assess a technical process by which Google can migrate the software components necessary to replicate the functionality of Google's Publisher Ad Server (DFP) and Ad Exchange (AdX) products to an acquirer's environment, and of the Final Auction Logic to an open source environment.[1]

3. In addition, I have been asked to describe the technical effort and time required to complete the AdX divestiture and the three phases of DFP divestiture remedies. This includes the timeline and resource estimates for migrating AdX to an acquirer's environment, creating application programming interfaces (APIs) within DFP that facilitate integration of Header Bidding wrappers and data export (phase 1 of DFP Divestiture), migrating the Final Auction Logic to an open source environment (phase 2 of DFP Divestiture), and migrating the remainder of DFP to an acquirer's environment (phase 3 of DFP Divestiture).

4. I understand that another technical expert retained by the Department of Justice in this matter, Dr. Jon Weissman, has analyzed the source code and data flow in AdX and DFP, how AdX and DFP are integrated with the rest of Google's products and services, how AdX and DFP are defined, and the technical feasibility of migrating the source code into an acquirer's software environment. I have reviewed his analysis and find it to be sound and reasonable. I accept his analysis as true for purposes of estimating the timelines and resources discussed in my report.

5. My opinions are based on my extensive experience managing, coordinating, and executing large-scale engineering efforts including mergers and acquisitions at large software companies, as well as analysis of publicly available material, internal documentation, emails, sworn

[1] I note that Google has repackaged AdX and DFP as "Google Ad Manager" or "GAM," but the functionality remains the same. I refer to AdX and DFP for consistency with the Plaintiffs' Revised Notice of Proposed Remedies, ECF No. 1482.

testimonies, presentations, and other documents produced in this case. A list of materials I have relied on is attached in Appendix A. If more information becomes available or I perform further analysis, I may modify or supplement my opinions.

6. I am being paid $400 per hour for my time preparing this report and in preparation for and during deposition and testimony. I received support from staff at Keystone Strategy, LLC. All the opinions and conclusions stated in this report are my own. My compensation does not depend upon the outcome of this matter or the substance of any opinion I reach.

II. QUALIFICATIONS

7. My academic background is in engineering and computer science, with a Ph.D. in Engineering Science from Clarkson University and an M.S. in Computer Science from Purdue University. My career includes over 25 years of professional experience across industry and academia focused on large-scale system reliability, availability, scalability, and performance (RASP), capacity engineering, and large-scale software testing. During my tenure in industry, I was responsible for tasks such as opening data centers in new regions, overseeing large migrations of software across data centers, mergers and acquisitions and accommodating the migration of the acquired software, and managing capacity inside and at the edges of private clouds. As part of this work, I coordinated with engineers necessary for each migration, estimated and scoped timelines for various workstreams, and ensured that capacity needs for the data centers (such as servers, networking, and power) were met. I have given presentations at conferences addressing these issues.

8. I have held senior technical roles at major technology companies including AT&T Labs, Network Appliance, Google, and Facebook. My roles were primarily as senior technical lead for performance, reliability and capacity teams responsible for predictive modeling, infrastructure validation, and cross-team integrations during service migrations and acquisitions.

9. I was employed at Facebook for over eight years, from 2010 to 2019, serving as the Lead Engineer for the Performance and Capacity team. My role involved managing the global expansion of data centers and overseeing the deployment of service and front-end clusters worldwide. In collaboration with production engineers and site operations teams. I worked

closely with individual engineering teams on significant software rewrites and major migrations, ensuring smooth integration into Facebook's infrastructure. I have worked on multiple migrations such as Instagram and WhatsApp's migrations into Facebook's infrastructure.

10. One project I spearheaded during my time at Facebook was project Lookback, where I developed strategic plans for establishing and staffing a dedicated team capable of meeting aggressive deadlines. Throughout my tenure, I frequently navigated complex challenges such as natural disaster recoveries for full data centers, providing decisive leadership during critical incidents and effectively managing responses to high-pressure situations to ensure timely resolution. My responsibilities also encompassed addressing various technical challenges, including network analysis, deployment analysis, performance analysis, and capacity analysis, which are all critical preparatory tasks for migrating AdX and/or DFP and establishing evaluation metrics to measure migration success. Additionally, my responsibilities included containerization, virtualization, and database migrations which will be critical to ensure the successful operation of AdX and/or DFP in the new target environment. I also executed numerous small- and medium-scale migrations, which the team responsible for divesting AdX and/or DFP will need to undertake.

11. I also worked at Google for over 5 years, from 2005 to 2010, where I was hired as a technical lead and asked to form the new Performance Testing team. In this role, I created performance testing benchmarks for a wide array of Google products and services including AdSense, AdWords, Writely (currently referred to as Google Docs), Trix (currently referred to as Google Spreadsheets), Preso (currently referred to as Google Slides) and almost all versions of Google Search (text, video, OneBoxes, etc.). During this time, I trained all new members of the Performance Team when they were hired, and I supervised multiple interns.

12. I have contributed to technical literature as an author and co-author of engineering textbooks, peer-reviewed papers, and have given numerous conference presentations on performance and capacity engineering. My work has consistently focused on enhancing the robustness and efficiency of software systems at scale, underpinned by rigorous testing and an engineering-driven mindset.

III. SUMMARY OF OPINIONS

13. I understand that Plaintiffs have proposed structural remedies where Google would divest parts of its Advertising Technology (AdTech) stack related to AdX and DFP.[2] I also understand that DFP divestiture would involve three phases, where Google is required to (1) develop APIs to share data and expand access to DFP, (2) open source the Final Auction Logic code,[3] and (3) divest the remainder of DFP if the first two phases were determined insufficient to restore competition.[4]

14. I describe the process by which Google can migrate software from its environment to an acquirer's environment or an open source environment in Sections VIII and IX. I explain how the described process follows standard industry procedures and matches my experience leading large-scale migrations and deployments. The process is also consistent with the process recommended by Google to its public cloud customers.

15. I have analyzed statistics summarizing Google's code repositories for DFP and AdX, [redacted]. In my expert opinion, AdX can be feasibly copied, migrated, and deployed by an acquirer within two years, and the Final Auction Logic can be feasibly copied, migrated, and deployed in open source environment within two years. I also conclude that if it is decided that a full divestiture of DFP should be pursued, then DFP can be feasibly migrated to and deployed by one or more acquirers within two years of the commencement of work after a divestiture order by the Court and identification of a buyer. With appropriate resources, these migrations can all be done with minimal disruption to customers and with

---

[2] Plaintiffs' Revised Notice of Proposed Remedies, pp. 8–11, ECF No. 1482.
[3] Throughout this report, I refer to the Final Auction Logic to mean the code that performs the final auction within the publisher ad server as defined in the Plaintiffs' Revised Notice of Proposed Remedies, pp. 9–11, ECF No. 1482. It is also consistent with the definition provided by Dr. Weissman in his report.
[4] Plaintiffs' Revised Notice of Proposed Remedies, pp. 9–11, ECF No. 1482.

minimal impact on the performance of these systems. I summarize the timelines for each component below.

16. From my analysis and review of Google's materials related to API integrations with additional AdTech tools, I believe it is feasible for Google to create the API integrations and data transfer APIs, as proposed by Plaintiffs, within 18 months after the Court issues a divestiture order in this case.

17. From my analysis and review of the timelines in Google's own documents, I believe it is feasible for Google to open source the Final Auction Logic that determines which ad to render on the page from available direct-sold opportunities and bids from indirect sources in a timely manner.[5] I conclude that this Final Auction can be operational and accessible through a party outside of Google's control within 24 months of when migration work commences after the Court issues a divestiture order and after the industry recipient(s) are identified. I believe it is feasible for this work to be done in parallel to the migration of AdX.

18. From my experience migrating large-scale software in industry, I believe it is feasible for Google to migrate AdX and the remainder of DFP as stipulated in Phase 3 of DFP divestiture within two years or less from when the migration work commences after a divestiture order by the Court and after a buyer is identified for each product. I present evidence that this migration would likely be made simpler by first migrating to Google's public cloud (Google Cloud Platform, or GCP). In my expert opinion, this can be done using processes that are common in companies with sophisticated networking systems that support high throughput, low-latency traffic.[6] The acquirer may then choose to use established industry practices to migrate the technical assets to their own software environment.

19. Figure III.1 presents a summary of the timetable of the analyses that I provide in detail in Sections VII, VIII, and IX.

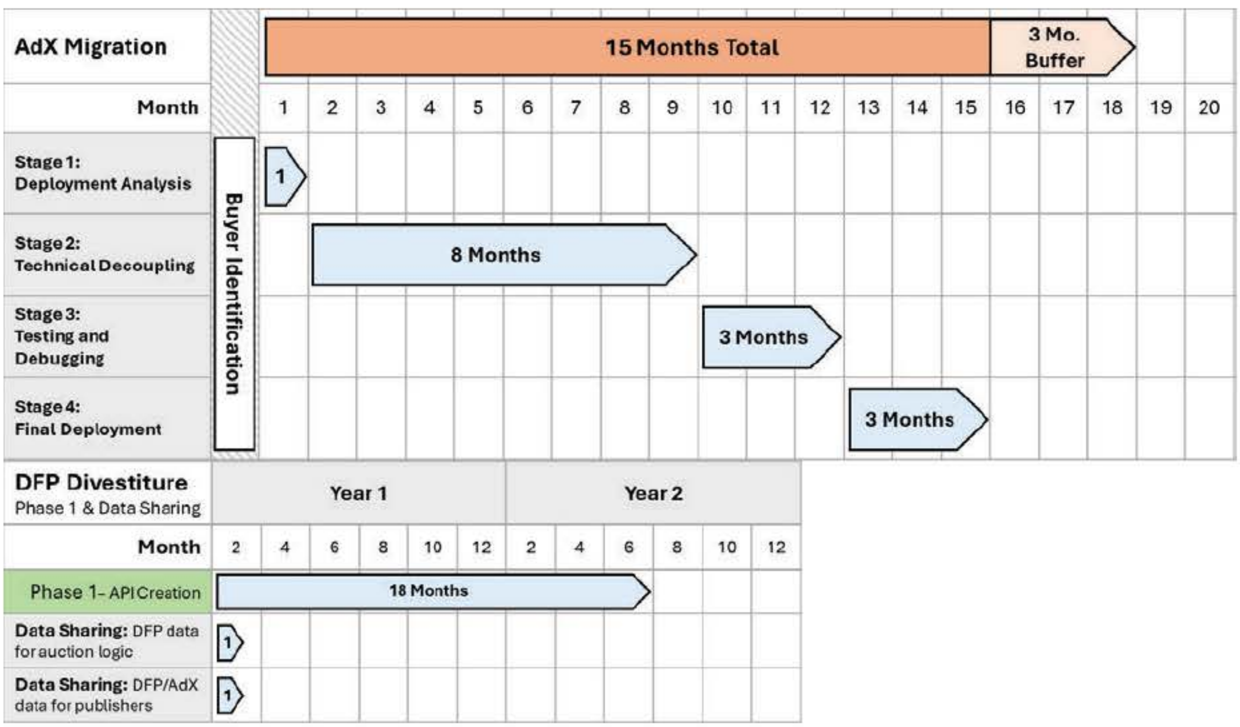
---

[5] Plaintiffs' Revised Notice of Proposed Remedies, pp. 9–11, ECF No. 1482.

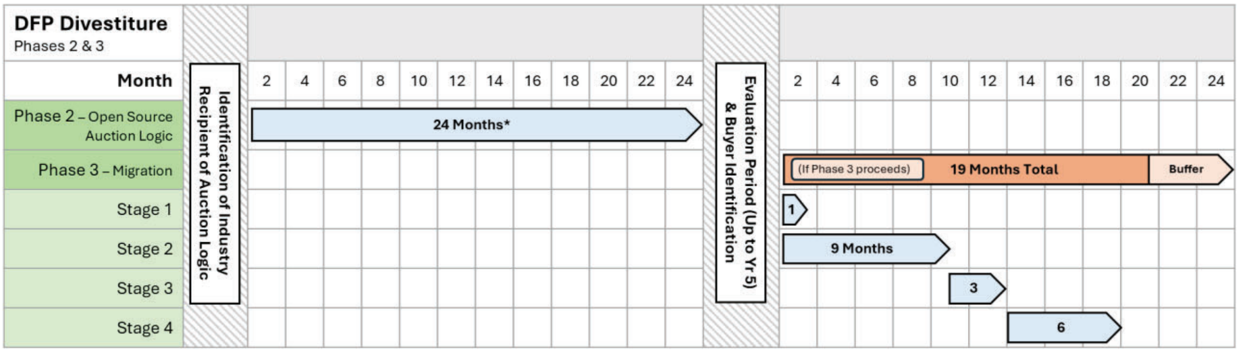
[6] I note that the advent of cloud computing services, which I address in more detail in Section IV below, makes such software integrations more efficient than they previously were, because cloud services obviate the need to transfer or set up physical assets such as storage devices.

20. I provide my analysis in three parts. First, I explain how software applications are deployed in a distributed manner in cloud environments, and I provide examples of how and why applications are routinely migrated between deployment environments. Second, I demonstrate how Google's current deployment environment that supports deployments of AdX and DFP applications adheres to industry standards that facilitate migration between software environments. I explain that Google operates a public cloud infrastructure (GCP) to which DFP and AdX can be initially migrated from Google's private infrastructure.[7] Finally, I provide a breakdown of the stages of the software migration process and present evidence to support an estimated timeline and resource requirements for each stage.

Figure III.1: Summary of Proposed Project Phases



[7] I discuss the details of this more in depth in Section IX. [redacted]. *See, e.g.*, GOOG-AT-MDL-B-009828764, at -797 ([redacted]).



*Some of the work in Phase 2 can begin prior to identification of the industry recipient.
Note that his timeline reflects my best estimate regarding the duration required to complete each part of the Plaintiffs' proposed remedies. For more details on how I came to these estimates, see Sections VII, VIII, and IX.

IV. ADX AND DFP ARE LARGE-SCALE SOFTWARE APPLICATIONS THAT CAN BE MIGRATED TO AN ACQUIRER'S ENVIRONMENT

21. Software applications are built from source code,[8] which is converted into machine-executable form and deployed to run on specific hardware by automated build and release pipelines.[9] Software applications execute on servers in data centers that rely on underlying operating systems and infrastructure to support the scale of deployments.

22. Google itself currently owns and operates its own distributed computing environment, more commonly referred to as a "cloud" computing environment.[10] Although currently deployed to Google's private cloud, in this report I describe how AdX and DFP can be migrated to Google's public cloud (GCP) and subsequently, to any other distributed system.

---

[8] "Source Code and Object Code," Office of Research, University of Washington, accessed July 3, 2025, https://www.washington.edu/research/glossary/source-code-and-object-code/ ("Source code is generally understood to mean programming statements that are created by a programmer with a text editor or a visual programming tool and then saved in a file.").
[9] *See* "Build and deploy automation and CI/CD tools," Google Cloud, last updated July 2, 2025, https://cloud.google.com/migration-center/docs/build-ci-cd-automation-tools ("The automation of building and deployment within a CI/CD framework is an essential part of the migration process."); "Deployment Methodology," Google Cloud, December 13, 2024, https://cloud.google.com/architecture/blueprints/enterprise-application-blueprint/deployment-methodology ("Pipelines provide a controllable, auditable, and repeatable mechanism for building out the blueprint.").
[10] *See* "What is distributed cloud?," IBM, accessed July 5, 2025, https://www.ibm.com/think/topics/distributed-cloud ("Distributed cloud is a public cloud computing service that lets you run public cloud infrastructure in multiple locations—your own cloud provider's data centers, other cloud providers' data centers, third-party data centers or colocation centers, and on-premises—manage everything from a single control plane."); What is Cloud Computing?," Google Cloud, accessed July 5, 2025, https://cloud.google.com/learn/what-is-cloud-computing ("Cloud computing is the on-demand availability of computing resources (such as storage and infrastructure), as services over the internet").

A. Like Many Large-Scale Software Systems, DFP and AdX are Deployed to Cloud Environments

23. AdX and DFP are software applications that are currently configured to run on Google's private cloud. In this section, I explain that AdX and DFP can be compiled and run on any compatible hardware environment.

1) The Cloud is a Distributed System of Servers that Run Software Applications

24. "The cloud" refers to on-demand availability of computing resources such as servers, storage, databases, and networking delivered over the internet from geographically distributed data centers.[11] Because these resources are pooled and redundant, cloud services remain highly available even if individual machines or sites fail, and they can elastically scale to handle massive numbers of simultaneous requests.[12]

25. It is common practice to deploy applications to cloud environments,[13] and software applications are commonly run on cloud infrastructure rather than on local or on-premises systems.[14]

a) Public Cloud Platforms Allow Organizations to Deploy Software Applications without Managing Proprietary or Internal Hardware

26. An acquirer of AdX or DFP could choose to deploy these applications on a public cloud of which there are several available. Organizations that offer cloud computing are known as

---

[11] *See* "What is Cloud Computing?," Google Cloud, accessed July 5, 2025, https://cloud.google.com/learn/what-is-cloud-computing ("Companies or individuals pay to access a virtual pool of shared resources, including compute, storage, and networking services, which are located on remote servers that are owned and managed by service providers.").

[12] *See* "Source Code and Object Code," Office of Research, University of Washington, accessed July 5, 2025, https://cloud.google.com/architecture/framework/reliability/build-highly-available-systems ("This principle in the reliability pillar of the Google Cloud Well-Architected Framework provides recommendations to plan, build, and manage resource redundancy, which can help you to avoid failures."); Stephanie Susnjara and Ian Smalley, "What is cloud computing?," IBM, February 10, 2025, https://www.ibm.com/think/topics/cloud-computing ("Cloud computing provides elasticity and self-service provisioning, so instead of purchasing excess capacity that sits unused during slow periods, you can scale capacity up and down in response to spikes and dips in traffic.").

[13] *See* "2025 State of the Cloud Report," Flexera, March 19, 2025, https://info.flexera.com/CM-REPORT-State-of-the-Cloud-2025-Thanks ("Cloud adoption has now reached the tipping point, with **over half of enterprise and SMB workloads currently running in public clouds**.") (emphasis in original).

[14] *See, e.g.*, "State of Financial Services in Cloud," Cloud Security Alliance, June 5, 2023, https://cloudsecurityalliance.org/artifacts/state-of-financial-services-in-cloud ("98% of respondents cited that their [financial] organization is using some form of cloud computing").

Cloud Service Providers (CSPs).[15] CSPs provide access to hardware that can dynamically scale compute resources at a global scale using highly automated distributed systems.[16] CSPs allow applications to be deployed on a scale that would otherwise be impractical to run on a single system. CSPs operate vast data centers around the world and offer services like storage, computing power, networking, and AI tools through the cloud.[17]

27. CSPs offer what is known as a "public" cloud.[18] That is, any organization can pay for the service to access cloud computing resources. Prior to availability of public clouds, every company needed its own hardware to run its applications and products. However, since the advent of commercial public cloud offerings in the last 15-20 years,[19] most companies have begun application deployment and public testing on a public cloud, and some continue to use the public cloud even after becoming very large, including successful enterprises like Netflix[20] and even Google itself.[21] This industry evolution means that complex systems like AdX and DFP no longer require a proprietary Google data center to run.

28. The ability to outsource software deployment and scaling allows organizations to focus on other software development priorities. CSPs can automatically scale up server instances during peak traffic periods, and shut them down during off-peak times, saving the organization

---

[15] Notable CSPs include Google Cloud Platform, Amazon Web Services, Microsoft Azure, and IBM Cloud.

[16] "Ensure operational readiness and performance using CloudOps," Google Cloud, accessed July 3, 2025, https://cloud.google.com/architecture/framework/operational-excellence/operational-readiness-and-performance-using-cloudops ("Autoscaling is an important strategy for dynamically adjusting your cloud resources based on workload fluctuations. By using autoscaling policies, you can automatically scale compute instances, storage, and other resources in response to changing demand.").

[17] *See* "What is Cloud Computing?," AWS, accessed June 24, 2025, https://aws.amazon.com/what-is-cloud-computing ("Cloud computing is the on-demand delivery of IT resources over the Internet with pay-as-you-go pricing. Instead of buying, owning, and maintaining physical data centers and servers, you can access technology services, such as computing power, storage, and databases, on an as-needed basis from a cloud provider like Amazon Web Services."); *see also* "Compare AWS and Azure services to Google Cloud," Google Cloud, December 3, 2024, https://cloud.google.com/docs/get-started/aws-azure-gcp-service-comparison (comparing services of major public cloud providers); *see also* Rehnuma Tasnim et al., "A Comparative Study on Three Selective Cloud Providers," *International Journal on Cybernetics & Informatics* 11, no. 4 (2022): 167–178, at 169–171, https://doi.org/10.5121/ijci.2022.110413 (providing a comparative study of Amazon, Azure and Digital Ocean's cloud services).

[18] *See* "What is cloud computing?," IBM, February 10, 2025, https://www.ibm.com/think/topics/cloud-computing ("A public cloud is a type of cloud computing in which a cloud service provider makes computing resources available to users over the public internet.").

[19] "Our Origins," AWS, accessed July 1, 2025, https://aws.amazon.com/about-aws/our-origins/ ("[W]e launched Amazon Web Services in the spring of 2006").

[20] *See* Section V.B.2.

[21] *See* Jordan Novet, "Google is moving parts of YouTube to its cloud service," CNBC, last modified June 4, 2021, https://www.cnbc.com/2021/06/04/google-is-moving-parts-of-youtube-to-its-cloud-service.html.

deploying the software from paying for extra servers when the extra servers are not needed.[22] When an organization needs to scale up application deployments (e.g., if they gain popularity in a new geographic location), this process is as simple as allocating more cloud resources through its CSP, rather than procuring new machines and building more infrastructure.[23]

29. There are many major public cloud providers. Google Cloud Platform, Amazon Web Services,[24] Microsoft Azure,[25] IBM Cloud,[26] and others each provide services to deploy and scale applications globally. These public clouds differ in specific features and pricing, but the core functionality of the providers is the same and they all enable an organization to run enterprise software at scale.

30. An acquirer of AdX or DFP could migrate these applications first to Google's platform, GCP, as the closest available public cloud to Google's private cloud,[27] to minimize differences in infrastructure with Google's private cloud. Once this is complete, the acquirer(s) could later choose to migrate the applications to any other public cloud provider, its own infrastructure,

---

[22] *See* Steren Giannini, "Cloud Run: Bringing serverless to containers," Google Cloud, May 20, 2019, https://cloud.google.com/blog/products/serverless/cloud-run-bringing-serverless-to-containers ("[W]ith Cloud Run, you can take an app—any stateless app—containerize it, and Cloud Run will provision it, scale it up and down"); "AWS Auto Scaling," AWS, accessed June 24, 2025 https://aws.amazon.com/autoscaling/ ("AWS Auto Scaling can help you optimize your utilization and cost efficiencies when consuming AWS services so you only pay for the resources you actually need. When demand drops, AWS Auto Scaling will automatically remove any excess resource capacity so you avoid overspending. AWS Auto Scaling is free to use, and allows you to optimize the costs of your AWS environment.").

[23] "AWS Auto Scaling," AWS, accessed June 24, 2025, https://aws.amazon.com/autoscaling/ ("AWS Auto Scaling automatically creates all of the scaling policies and sets targets for you based on your preference. AWS Auto Scaling monitors your application and automatically adds or removes capacity from your resource groups in real-time as demands change.").

[24] "AWS Global Infrastructure," AWS, accessed July 3, 2025, https://aws.amazon.com/about-aws/global-infrastructure/ ("The AWS Global Cloud Infrastructure is a secure, extensive, and reliable cloud infrastructure, offering over 200 fully featured services from data centers globally.").

[25] "Azure App Service," Microsoft Azure, accessed July 3, 2025, https://azure.microsoft.com/en-us/products/app-service/ ("Quickly build, deploy, and scale web apps and APIs globally with a fully managed platform as a service (PaaS).").

[26] "Cloud Infrastructure Solutions," IBM, accessed July 3, 2025, https://www.ibm.com/cloud/infrastructure ("IBM Cloud® is constantly expanding its global footprint to help you meet your customers where they need you. Deploy, control and manage workloads in more than 60 data centers and six multizone regions with availability zones.").

[27] "Google Cloud infrastructure," Google Cloud, accessed July 5, 2025, https://cloud.google.com/infrastructure ("With a highly provisioned, low-latency network—the same network that powers products like Gmail, Google Search, and YouTube—your traffic stays on Google's private backbone for most of its journey, ensuring exceptional user experience and high performance.").

or a combination of the two as the environment in which to host the applications. Migrating from GCP to another cloud service provider is a well-established practice.[28]

b) Private Cloud, or Internal Data Centers, Allow Organizations Additional Customizability When Deploying Large-Scale Software Applications

31. Google utilizes a private cloud for enhanced security, control over data, compliance with regulatory requirements, and performance optimization, and an acquirer can choose to deploy AdX or DFP that way, after the shift. Some global organizations reach the point at which it becomes beneficial to run their own private cloud with infrastructure dedicated exclusively to their own organizations.[29]

32. Google's public cloud offers many of these tools as customizable options for large enterprises. For example, should an organization choose to deploy a portion of its software on GCP, tools like Google's Anthos (now a part of Google Distributed Cloud[30]) let the organization deploy containerized workloads across both on-premises and public cloud environments.[31]

33. Regardless of whether an acquirer of DFP and AdX chooses to deploy on a public cloud or its own private cloud, it should typically be able to deploy the software using standard server-grade hardware, with some effort to account for differences in performance and design.[32] There are some applications that can benefit from custom

---

[28] *See, e.g.*, Ramy Afifi, "Migrate compute from Google Cloud Platform (GCP) to AWS using AWS Application Migration Service," AWS, September 28, 2023, https://aws.amazon.com/blogs/storage/migrate-from-google-cloud-platform-gcp-to-aws-using-aws-application-migration-service/ ("Customers using Google Cloud Platform (GCP) might explore the option of spreading or transitioning their cloud usage away from GCP to alternative providers for various reasons, including cost evaluations, data centralization, or changes in business requirements.").
[29] Mariusz Michalowski, "55 Cloud Computing Statistics for 2025," Spacelift, last modified January 1, 2025, https://spacelift.io/blog/cloud-computing-statistics#public-vs-private-cloud-statistics ("84% of companies use at least one private cloud" while "96% of companies are using at least one public cloud").
[30] "Google Distributed Cloud (software only) for bare metal overview," Google Cloud, accessed July 2, 2025, https://cloud.google.com/kubernetes-engine/distributed-cloud/bare-metal/docs/concepts/about-bare-metal ("We also offer Google Distributed Cloud as a software-only product …previously known as Anthos clusters on bare metal").
[31] *See* Jeff Reed and Chen Goldberg, "Introducing Anthos for VMs and tools to simplify the developer experience," Google Cloud, October 13, 2021, https://cloud.google.com/blog/topics/hybrid-cloud/introducing-anthos-for-vms-and-other-app-modernization-tools ("Anthos unifies the management of infrastructure and applications across on-premises, edge, and multiple public clouds, as well as ensuring consistent operation at scale.").
[32] *See* "Machine families resource and comparison guide," Google Cloud, last updated July 3, 2025, https://cloud.google.com/compute/docs/machine-resource (describing and comparing various machine families, machine series, and machine types that could be used).

hardware, such as training AI models.[33] However, even in that case, recent DeepSeek results have shown that it is possible for standard server-grade hardware to not only to match the performance of specialized hardware, but to surpass it.[34] Furthermore, some of Google's specialized hardware, such as its Tensor Processing Units, are available to users of GCP, meaning that an acquirer could utilize these offerings if the acquirer determines them to be beneficial.[35]

2) Cloud Environments Facilitate Large-Scale Deployment of Software Applications, such as AdX and DFP

34. As discussed in Section IV.A.1, cloud environments offer significant benefits when deploying large-scale software applications. In this section, I discuss why AdX and DFP, which currently run on Google's private cloud platform, can run on public cloud platforms.

a) Cloud Platforms Provide Redundancy in the Case of Hardware Failure

35. One major benefit of cloud platforms for large-scale deployments is their reliability due to redundancy. Large-scale applications, including Google's current AdTech stack, are deployed across multiple locations and have copies of data stored at multiple locations.[36] This means that, for example, if a server processing traffic for AdX or DFP experiences an outage, there are multiple resources in place to minimize disruption in service.[37] CSPs provide tools and

---

[33] "Accelerate AI development with Google Cloud TPUs," Google Cloud, accessed July 3, 2025, https://cloud.google.com/tpu ("[TPUs] are optimized for training and inference of AI models.").

[34] *See* Mojahid Mottakin, "Putting DeepSeek to the test: how its performance compares against other AI tools," The Conversation, February 4, 2025, https://theconversation.com/putting-deepseek-to-the-test-how-its-performance-compares-against-other-ai-tools-248368 ("DeepSeek's success is already challenging the status quo, demonstrating that high-performance LLM models can be developed without billion-dollar budgets.").

[35] *See* "Accelerate AI development with Google Cloud TPUs," Google Cloud, accessed June 24, 2025, https://cloud.google.com/tpu?hl=en ("What is a Tensor Processing Unit (TPU)? Google Cloud TPUs are custom-designed AI accelerators, which are optimized for training and inference of AI models."); "Manage TPU resources," Google Cloud, last modified July 2, 2025, https://cloud.google.com/tpu/docs/managing-tpus-tpu-vm#create-node-api ("You can create a Cloud TPU using gcloud, the Google Cloud console, or the Cloud TPU API.").

[36] Ashish Gupta and Jeff Shute, "High-Availability at Massive Scale: Building Google's Data Infrastructure for Ads," Google Inc., accessed July 3, 2025, https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/44686.pdf. ("As part of Google's advertising infrastructure, we have built and run several large-scale streaming systems … and have learned that building natively multi-homed systems is the best solution in most cases.").

[37] "Building blocks of reliability in Google Cloud," Google Cloud, November 20, 2024, https://cloud.google.com/architecture/infra-reliability-guide/building-blocks ("Google Cloud infrastructure is designed to tolerate and recover from failures.…[Regional resources are deployed] with the default (multi-zone) configuration").

systems for managing these redundancies across different regions.[38] This provides the high degree of reliability that is necessary for large-scale applications such as AdX and DFP.

36. Organizations often deploy software applications using standard redundancy strategies.[39] For example, Meta's systems auto-detect failures and remediate them at scale, keeping services healthy during live tests.[40]

b) Cloud Platforms Provide Scalability

37. Public cloud platforms can reduce latency and increase throughput by adding computational power where and when it is required.[41] For instance, GCP's Compute Engine's "managed instance groups" feature offers autoscaling capabilities based on load capacity, ensuring services maintain consistent performance.[42] As more users access the service, more capacity is

[38] "Architecting disaster recovery for cloud infrastructure outages," Google Cloud, last modified May 10, 2024, https://cloud.google.com/architecture/disaster-recovery ("Google designs its infrastructure to meet aggressive availability targets based on our extensive experience building and running modern data centers. Google is a world leader in data center design. From power to cooling to networks, each data center technology has its own redundancies and mitigations, including FMEA plans. Google's data centers are built in a way that balances these many different risks and presents to customers a consistent expected level of availability for Google Cloud products. Google uses its experience to model the availability of the overall physical and logical system architecture to ensure that the data center design meets expectations.").

[39] "Implementing Hardware Redundancy," High-Availability, accessed July 6, 2025, https://www.high-availability.com/articles/system/hardware-redundancy ("As the system is only as strong as it's weakest component, the availability levels of most hardware components in a system should be increased by using high-quality and proven enterprise-grade products together with the introduction of system redundancy. This means the introduction of multiple components or additional capacity to ensure uninterrupted service so that a localised fault in that component can be tolerated resulting in no system downtime. Higher levels of availability are achieved by reducing or removing Single Points of Failure in the system. Examples of hardware redundancy include: Dual power supplies[,] Multiple network cards[,] RAID storage[,] Cooling fans[,] Multiple storage (multipath) connections").

[40] Gerald Guo and Thawan Kooburat, "Scaling services with Shard Manager," Engineering at Meta, August 24, 2020, https://engineering.fb.com/2020/08/24/production-engineering/scaling-services-with-shard-manager/ ("Automatic failure detection and shard failover: Shard Manager can automatically detect server failures and network partition."); Fred Lin, Harish Dattatraya Dixit, and Sriram Sankar, "How Facebook keeps its large-scale infrastructure hardware up and running," Engineering at Meta, December 9, 2020, https://engineering.fb.com/2020/12/09/data-center-engineering/how-facebook-keeps-its-large-scale-infrastructure-hardware-up-and-running/ ("[W]e automate root cause analysis for hardware and system failures at scale to get to the bottom of issues quickly.").

[41] "Load balancing and scaling," Google Cloud, last updated June 13, 2025, https://cloud.google.com/compute/docs/load-balancing-and-autoscaling ("Compute Engine offers autoscaling to automatically add or remove VM instances from a managed instance group (MIG) based on increases or decreases in load. Autoscaling lets your apps gracefully handle increases in traffic, and it reduces cost when the need for resources is lower.").

[42] "Load balancing and scaling," Google Cloud, last updated July 2, 2025, https://cloud.google.com/compute/docs/load-balancing-and-autoscaling ("Compute Engine offers autoscaling to automatically add or remove VM instances from a managed instance group (MIG) based on increases or decreases in load."); Cloud Architecture Center, "Patterns for Scalable and Resilient Apps," Google Cloud, last reviewed May

available to meet the demand. In that sense, the complexity of the software application is not strictly related to the size of the deployment.

38. Scalability is important because an acquirer operating AdX, DFP, or the open source auction will need to implement elastic scaling policies. This flexibility enables the system to scale up during peak bidding periods and scale down during lulls, ensuring high availability and cost efficiency.[43]

39. Additionally, cloud environments can support any storage and data requirements. Platforms like GCP's BigQuery support multi-petabyte storage, with internal systems routinely managing dozens of petabytes across data.[44] AWS offers similar scale, with services like S3 handling hundreds of trillions of objects, and physical data import tools (Snowball[45] and Snowmobile[46]) transferring multiple petabyte of data securely.[47]

c) Cloud Platforms Allow Location and Hardware Configurability

40. Outside of Google's private infrastructure, AdX and DFP can still be deployed and scaled globally. GCP operates in over 40 regions (a specific geographic area where GCP resources

5, 2025, https://cloud.google.com/architecture/scalable-and-resilient-apps?hl=en ("Scalability is the measure of a system's ability to handle varying amounts of work by adding or removing resources from the system… The flexibility to adjust the resources consumed by an app is a key business driver for moving to the cloud… Google Cloud provides products and features to help you build scalable, efficient apps… Autoscaling helps you to automatically scale the computing resources consumed by your app.").

[43] "Manage traffic and load for your workloads in Google Cloud," Google Cloud, last reviewed November 20, 2024, https://cloud.google.com/architecture/infra-reliability-guide/traffic-load ("When the load on your application increases, autoscaling helps to improve the availability of the application by provisioning additional resources automatically. During periods of low load, autoscaling removes unused resources, and helps to reduce cost").

[44] Gerrit Kazmaier, "Google Cloud Data Analytics 2021: The year in review," Google Cloud, December 22, 2021, https://cloud.google.com/blog/products/data-analytics/biggest-moments-for-google-clouds-unified-data-cloud-2021 ("[Verizon] ingest[s] 200TB Daily, store[s] 100PB in BigQuery, [and] stream[s] 300MB per second… [m]ore than 3,000 organizations shared more than 250 petabytes of data…").

[45] "AWS Snowball," AWS, accessed July 6, 2025, https://aws.amazon.com/snowball/ ("Migrate data at petabyte-scale[:] Move databases, backups, archives, healthcare records, analytics datasets, IoT sensor data and media content to the cloud - especially when network conditions are limited.").

[46] "AWS Snowmobile: Migrate Data to the Cloud With the World's Biggest Hard Disk," NetApp BlueXP, January 3, 2022, https://bluexp.netapp.com/blog/aws-cvo-blg-aws-snowmobile-migrate-data-with-worlds-biggest-hard-disk ("AWS Snowmobile is a data transfer service designed to move very large amounts of data, at the Exabyte scale, to Amazon storage services.").

[47] "Large data migration using AWS Snowball Edge," AWS, accessed July 6, 2025, https://docs.aws.amazon.com/snowball/latest/developer-guide/LargeDataMigration.html ("Large data migration from on-premises locations requires careful planning, orchestration, and execution to ensure that your data is successfully migrated to AWS.").

are deployed) and 120 zones (a location within a region) worldwide.[48] Cloud platforms such as GCP allow providers to place services very close to end-users, significantly reducing latency.[49] This is a key benefit of globally distributed data centers.

d) Cloud Platforms Allow Interoperability Between Different Software and Hardware

41. Modern cloud systems allow the same software to run on different hardware.[50] This means that when AdX and DFP are migrated, they do not need to run on the same hardware that they currently run on. This flexibility is achieved using a technique called containerization, which I discuss more in depth in Section IV.B.1. For example, AWS's SageMaker enables developers to build, train, and deploy machine learning models efficiently between types of processors. It is capable of transitioning workloads from traditional x86 processors to AWS's ARM-based Graviton processors without code modifications.

B. Engineers Routinely Migrate Large-Scale Deployments of Software Applications Within and Between Cloud Environments Without Customer Disruption

42. Engineers frequently migrate, upgrade, and reconfigure production systems while those systems are live, often without users noticing any changes at all. These migrations are a standard part of operating large-scale infrastructure; organizations like Google specifically design their systems with principles like scalability and operational independence in mind from the start.[51]

---

[48] "Cloud locations," Google Cloud, accessed June 24, 2025, https://cloud.google.com/about/locations/.
[49] *See* Rodrigo S. Couto et al., "Latency Versus Survivability in Geo-Distributed Data Center Design," IEEE Global Communications Conference (2014), at 1 ("Indeed, the current trend in data center (DC) networking is to create a geographically-distributed architecture spanning a few sites across wide area networks. This allows to reduce the Cloud access latency to end users"); "Geography and regions," Google Cloud, last updated July 2, 2025, https://cloud.google.com/docs/geography-and-regions ("Google Cloud infrastructure services are available in locations across North America, South America, Europe, Asia, the Middle East, and Australia. These locations are divided into regions and zones. You can choose where to locate your applications to meet your latency, availability, and durability requirements."); *see also* Fabio Palumbo et al., "Characterization and Analysis of Cloud-to-User Latency: the case of Azure and AWS," *Computer Networks* 184, no. 4 (2020).
[50] "What is Containerization?," AWS, accessed July 7, 2025, https://aws.amazon.com/what-is/containerization/ ("Traditionally, to run any application on your computer, you had to install the version that matched your machine's operating system... with containerization, you can create a single software package, or container, that runs on all types of devices and operating systems.").
[51] Omkar Suram and Rob Rosen, "Optimize your system design using Architecture Framework Principles," Google Cloud, December 15, 2021, https://cloud.google.com/blog/topics/solutions-how-tos/optimize-your-system-design-

43. At companies like Google, software migrations occur frequently. There has not been a month and in many cases, not even a week when a Google engineering team is not deploying new software to its servers.[52] These internal migrations include transitions between clusters, regions, data centers, storage systems, network stacks, and more and can be done without interrupting service to the users.[53] Google's infrastructure is designed to make such changes seamless and resilient, even under demanding latency and throughput requirements.[54] Google's expertise in software migration enables the smooth migration of AdX and DFP.

44. In Section VI.C, I discuss how Google's monitoring of AdX and DFP systems facilitate assessments necessary for migration.

1) <u>There Are Well-Established Processes to Migrate Deployments of Software Applications</u>

45. The advent of cloud computing and migration techniques makes it possible and efficient in most instances to transfer ownership and control of software applications without transferring physical hardware, as an acquiring company can leverage cloud environments as discussed in Section IV.A.

46. Software applications operate in particular software environments.[55] A software environment encompasses all of the tools, libraries, configurations, and dependencies needed to run an application. Typical examples include operating systems (Linux or Windows), middleware

---

using-architecture-framework-principles ("A robust system design is secure, reliable, scalable, and independent, enabling you to apply changes atomically, minimize potential risks, and improve operational efficiency.").

[52] Deposition of Nitish Korula (Google), June 9, 2025, 102:3–105:11 ("Q. And approximately how many times were you involved, or your team involved, in overseeing the deployment of new software onto Google's [ads group] systems? A.…I would say multiple times a week, we would deploy new code…many dozens to hundreds of changes that [the engineers] would submit and that would get deployed every year… a particular server was deployed, every so often, say, twice a week…[after testing was successful] it would keep going until eventually it was deployed to all of our data centers…this is the case for like the smallest of changes.").

[53] "Live migration process during maintenance events," Google Cloud, last updated July 2, 2025, https://cloud.google.com/compute/docs/instances/live-migration-process ("Live migration lets Google Cloud perform maintenance without interrupting a workload, rebooting an instance, or modifying any of the instance's properties, such as IP addresses, metadata, block storage data, application state, or network settings.… Google Cloud copies all the guest memory to the target, while tracking the pages that have been changed on the source.").

[54] "Designing resilient systems," Google Cloud, last updated July 2, 2025, https://cloud.google.com/compute/docs/tutorials/robustsystems.

[55] "About service execution environments," Google Cloud, last updated July 2, 2025, https://cloud.google.com/run/docs/about-execution-environments ("*Cloud Run services* by default don't have an execution environment specified, which means Cloud Run selects the execution environment based on the features used.") (emphasis in original).

(Apache web servers, MySQL databases), and runtime libraries (e.g., Python or Java frameworks). Migrating software involves replicating or reconfiguring the entire environment to maintain consistent functionality.[56]

47. Modern software applications are intentionally designed to be environment-agnostic.[57] Google itself has tools like Anthos that enable a user to operate under multiple cloud environments, acknowledging the value of utilizing multiple cloud platforms' services.[58] Engineers minimize direct hardware or platform dependencies when building new software, anticipating future migrations.[59] The main practice for achieving this is containerization, where technologies like Docker and Kubernetes bundle applications with all of their required components. Google has been running containerized workloads in production for more than a decade. Google itself pioneered container use at massive scale and built Kubernetes to orchestrate this process in the same environments where cloud migration occurs.[60] Container orchestration tools like GKE and Docker ensure that differences between physical servers and virtual environments become unimportant for the application's behavior.[61]

---

[56] "Replicate assets in a cloud migration," Microsoft, April 10, 2024, https://learn.microsoft.com/en-us/azure/cloud-adoption-framework/migrate/deploy/replicate ("In a cloud migration, you replicate and synchronize assets over a network between an existing datacenter and the cloud.").

[57] Rajkumar Buyya, Christian Vecchiola, and S. Thamarai Selvi, *Mastering Cloud Computing: Foundations and Applications Programming* (Waltham, MA: Morgan Kaufmann, 2013), 20 ("any piece of code that performs a task can be turned into a service… A service is supposed to be *loosely coupled*, *reusable*, *programming language independent*, and *location transparent*.") (emphasis in original).

[58] Alok Jain and Ronen Kofman, "How Anthos supports your multicloud needs from day one," Google Cloud, April 29, 2021, https://cloud.google.com/blog/topics/anthos/how-anthos-makes-multicloud-possible-today ("[T]he ability to run workloads on multiple cloud providers is becoming increasingly important. Why? There are multiple reasons. Some organizations want application teams to be able to take advantage of the best service for a given application. Others have acquired a company which runs on another cloud.").

[59] "Portability," MicrosoftEngineering Fundamentals Playbook, last modified August 26, 2024, https://microsoft.github.io/code-with-engineering-playbook/non-functional-requirements/portability/ ("Portability refers to the ease with which software can be transferred and used in different environments or platforms without requiring significant modification. This includes moving the software across various hardware, operating systems, cloud services, or development frameworks while maintaining its functionality, performance, and usability.").

[60] "GKE overview," Google Cloud, last updated July 2, 2025, https://cloud.google.com/kubernetes-engine/docs/concepts/kubernetes-engine-overview ("Kubernetes was developed by Google, drawing on years of experience operating production workloads at scale on Borg, our in-house cluster management system. With GKE, you can deploy and operate your own containerized applications at scale using Google's infrastructure.").

[61] "What is Container Orchestration?," AWS, accessed June 24, 2025, https://aws.amazon.com/what-is/container-orchestration/ ("Containerization bundles an application's code with all the files and libraries it needs to run on any infrastructure…Containers have become the standard unit of computing for cloud-native applications.…The only requirement to be able to run containers is that the server itself runs a containerization service like Docker. Docker is an open source tool for packaging software and associated libraries, system tools, code, and runtime into a container.… [Container orchestration tools] Run different versions of applications (for example, test and production across CI/CD) at once.").

48. Software applications are often divided into discrete "tasks" or "microservices" that can be deployed to any suitable environment.[62] By treating tasks as portable units of work, companies can reroute software components across different environments without changing their functionality.[63] Modern orchestration platforms allow these tasks to be automatically scheduled on available servers. For example, Google's Borg cluster manager runs hundreds of thousands of tasks across tens of thousands of machines, automatically distributing work.[64]

49. The software migration process consists of multiple steps, which I elaborate on in detail with respect to the migration of AdX and DFP in Section IX.

2) There Are Many Reasons to Migrate Software Deployments Between Environments

50. Companies migrate software for strategic reasons, including technological upgrades, improved scalability, performance enhancements, cost optimization, regulatory compliance, and geographic distribution needs.[65] Google itself has performed these types of large-scale migrations in the past, such as when it announced moving parts of its YouTube service from Google's internal data centers to the Google Cloud Platform in 2021 to capitalize on its cloud

[62] Davide Taibi, Valentina Lenarduzzi, and Claus Pahl, "Continuous Architecting with Microservices and DevOps: A Systematic Mapping Study," Cloud Computing and Services Science. CLOSER 2018 Selected papers. *Communications in Computer and Information Science* 1073 (2019): 126–151, at 127 ("Microservices are relatively small and autonomous services deployed independently, with a single and clearly defined purpose. Because of their independent deployability, they have a lot of advantages for continuous delivery. They can be developed in different programming languages, they can scale independently from other services, and they can be deployed on the hardware that best suits their needs. Moreover, because of their size, they are easier to maintain and more fault-tolerant since the failure of one service will not break the whole system, which could happen in a monolithic system.").

[63] Gaurav Shekhar, "The Role of Containers and Orchestration in Scalable System Design," *ESP Journal of Engineering & Technology Advancements* 3, no. 4 (2023): 61–72, at 63 ("Containers encapsulate an application and all of its dependencies—such as libraries and configuration files—into a single, self-contained unit. This encapsulation ensures that the application runs consistently across different environments, whether it is a developer's laptop, a test server, or a production cloud environment. As a result,containers eliminate the 'it works on my machine' problem, providing a reliable and repeatable way to deploy applications. This portability is crucial in multi-cloud and hybrid cloud environments, where applications may need to be moved or run across various platforms and infrastructures.").

[64] Abhishek Verma et al., "Large-scale cluster management at Google with Borg," *Proceedings of the European Conference on Computer Systems, ACM,* no. 18 (2015): 1–17, at 1 ("Google's Borg system is a cluster manager that runs hundreds of thousands of jobs, from many thousands of different applications, across a number of clusters each with up to tens of thousands of machines").

[65] *See* Section V for examples; Tom Nikl and Ravi Kiran Chintalapudi, "8 Common Reasons Why Enterprises Migrate to the Cloud," Google Cloud, November 28, 2018, https://cloud.google.com/blog/products/storage-data-transfer/8-common-reasons-why-enterprises-migrate-to-the-cloud.

4) Many Tools are Available to Assess Migration Complexity of Software Application Deployments

74. A broad array of tooling is available to assess the complexity of migrating software systems like AdX and DFP. Google itself offers a wide variety of tools, such as Network Analyzer, to monitor network configurations,[99] Connectivity Tests, to simulate packet flows over networks,[100] the Database Migration Assessment (DMA) tool, to evaluate database schemas and estimate effort of a migration,[101] and the Flow Analyzer, to evaluate traffic patterns.[102] This list is by no means exhaustive, and there are also numerous tools outside of GCP for this purpose.

75. Distributed tracing systems, such as OpenTelemetry and Zipkin, can be instrumented to track request paths end-to-end across microservices. Google's own services are capable of tracing and structured logging through Dapper, a production tracing tool that records the path of requests through microservices to provide developers with a detailed "trace" of what each service call did and how long it took. This real-time monitoring is vital for understanding how services will behave post-migration and identifying what changes, if any, are required to maintain functionality.[103]

Michael E. Bernstein, *Software Engineering: Modern Approaches*, 2nd ed. (Long Grove: Waveland Press, Inc., 2011), 145 ("Experience of the authors and others shows that the number of developers with whom each developer needs to interact on a regular basis should normally be between three and seven. (Humphrey [3] suggests four to eight.)").

[99] Manasa Chalasani and Mary Colley, "Introducing Network Analyzer: One stop shop to detect service and network issues," Google Cloud, May 26, 2022, https://cloud.google.com/blog/products/networking/introducing-network-analyzer-detect-service-and-network-issues ("Network Analyzer offers an out-of-the-box suite of always-on analyzers that continuously monitor GCE and GKE network configuration.").

[100] "Connectivity Tests overview," Google Cloud, last updated July 2, 2025, https://cloud.google.com/network-intelligence-center/docs/connectivity-tests/concepts/overview ("Connectivity Tests is a diagnostics tool that lets you check connectivity between network endpoints. It analyzes your configuration and, in some cases, performs live data plane analysis between the endpoints.").

[101] Celia Antonio and Pritesh Jani, "Save time, money and modernize your legacy database estate .... but first assess," Google Cloud, July 12, 2023, https://cloud.google.com/blog/products/databases/introducing-open-source-database-migration-assessment-tool ("The Database Migration Assessment (DMA) is a no-cost customer engagement in which we collect database metadata and deliver a detailed customized customer readout.…Once the metadata is collected, it gets processed and our database expert team delivers a detailed customized readout containing… Complete migration plan").

[102] "Monitor your traffic flows," Google Cloud, last updated July 2, 2025, https://cloud.google.com/network-intelligence-center/docs/flow-analyzer/monitor-traffic-flows.

[103] *See* Benjamin H. Sigelman et al., "Dapper, a Large-Scale Distributed Systems Tracing Infrastructure," Google Technical Report, April 2010, https://netman.aiops.org/~peidan/ANM2019/7.TraceAnomalyDetection/ReadingList/2010Google_Dapper.pdf ("Dapper is deployed across virtually all of Google's systems, and has allowed the vast majority of our largest

76. The technology for system migrations has evolved rapidly. Many of the tools now considered standard in system migrations were simply not available until recently. For example, Network Analyzer was released in 2022,[104] Connectivity Tests in 2019,[105] DMA in 2023,[106] Flow Analyzer in 2025,[107] OpenTelemetry in 2019,[108] Zipkin in 2012, and Dapper in 2010.[109] Today, engineers benefit from automated dependency graphing tools, schema analyzers, and synthetic testing frameworks. The challenge of migrating complex systems like AdX or DFP follows a standardized path. Tools like Bazel (Google's own build system),[110] Visor (a container sandbox),[111] and Config Validator (for policy enforcement)[112] provide assurance of compatibility. These advancements make migration not only feasible but routine for experienced teams.

---

workloads to be traced without need for any application-level modifications, and with no noticeable performance impact. Dapper's utility to developers and operations teams is evidenced by the popularity of the main tracing user interface").

[104] Manasa Chalasani and Mary Colley, "Introducing Network Analyzer: One stop shop to detect service and network issues," Google Cloud, May 26, 2022, https://cloud.google.com/blog/products/networking/introducing-network-analyzer-detect-service-and-network-issues.

[105] Tim Anderson, "Google emits Network Intelligence Center to help untangle misconfigured cloud networks," The Register, November 14, 2019, https://www.theregister.com/2019/11/14/google_network_intelligence_center_for_cloud_and_onpremises/.

[106] Celia Antonio and Pritesh Jani, "Save time, money and modernize your legacy database estate …. but first assess," Google Cloud, July 12, 2023, https://cloud.google.com/blog/products/databases/introducing-open-source-database-migration-assessment-tool/.

[107] "Network Intelligence Center release notes," Google Cloud, last updated July 2, 2025, https://cloud.google.com/network-intelligence-center/docs/release-notes ("February 27, 2025[:] Flow Analyzer is available in **General availability**.") (emphasis in original).

[108] "OpenTelemetry," Cloud Native Computing Foundation, accessed July 6, 2025, https://www.cncf.io/projects/opentelemetry/ ("OpenTelemetry was accepted to CNCF on May 7, 2019 and moved to the **Incubating** maturity level on August 26, 2021.") (emphasis in original).

[109] John Willis, "A History of Distributed Tracing," DevOps, December 6, 2022, https://devops.com/a-history-of-distributed-tracing/ ("Dapper…was introduced by Google in 2010. Two years after…Twitter open sourced Zipkin").

[110] "Review the dependency graph," Bazel, last updated July 4, 2025, https://bazel.build/tutorials/cpp-dependency ("Bazel uses those statements to create the project's dependency graph, which enables accurate incremental builds.").

[111] Nicolas Lacasse, "Open-sourcing gVisor, a sandboxed container runtime," Google Cloud, May 2, 2018, https://cloud.google.com/blog/products/identity-security/open-sourcing-gvisor-a-sandboxed-container-runtime ("To that end, we'd like to introduce gVisor, a new kind of sandbox that helps provide secure isolation for containers, while being more lightweight than a virtual machine (VM). gVisor integrates with Docker and Kubernetes, making it simple and easy to run sandboxed containers in production environments.").

[112] Adrien Walkowiak, "Protecting your GCP infrastructure at scale with Forseti Config Validator," Google Cloud, September 17, 2019, https://cloud.google.com/blog/products/identity-security/protecting-your-gcp-infrastructure-at-scale-with-forseti-config-validator ("The goal is to write your security policies as code once and for all, and to apply them both before and after you deploy resources in your GCP environment.").

V. INDUSTRY EXAMPLES DEMONSTRATE SUCCESSFUL MIGRATION OF COMPLEX LARGE-SCALE SOFTWARE APPLICATIONS ACROSS DIFFERENT ENVIRONMENTS

77. As discussed in Section IV.B, software migration is common across all software industries. No two migrations are the same, but engineers approach them with a standardized set of procedures and tools. There are many industry examples of software migrations at similar scale to the migrations at issue in this case. In this section, I discuss examples of software migration in industry due to both voluntary acquisitions and regulatory divestitures. I also discuss examples of software migrations across cloud environments. These examples illustrate that complex, data-rich, high-availability applications can be migrated efficiently and reliably and that the separation of AdX and DFP would not be unprecedented migrations.

A. Software Application Deployments Have Been Successfully Migrated Following Corporate Acquisitions and Divestitures, Similar to that of a Divestiture of AdX or DFP

78. These examples show that large-scale consumer software platforms with significant media and user data can be absorbed into new ownership and infrastructure environments while continuing to operate without disruption.

1) Facebook Acquisition of Instagram

79. When Facebook acquired Instagram in 2012, it dedicated a small team of engineers to work on migrating the software from Instagram's infrastructure to Facebook's.[113] This team of eight engineers grew to 20 during the "Instagration." The migration of Instagram happened without service interruptions, often called a "live migration." In about one year, Instagram was migrated from an Amazon cloud to Facebook's private data center.[114]

80. In order to transfer Instagram to Facebook's private servers, the team of engineers first moved Instagram's code into Amazon's Virtual Private Cloud (VPC).[115] This was a useful step because

---

[113] Evelyn M. Rusli, "Facebook Buys Instagram for $1 Billion," New York Times, April 9, 2012, https://archive.nytimes.com/dealbook.nytimes.com/2012/04/09/facebook-buys-instagram-for-1-billion/.
[114] Cade Metz, "How Facebook Moved 20 Billion Instagram Photos Without You Noticing," Wired, June 26, 2014, https://www.wired.com/2014/06/facebook-instagram/.
[115] *See* Cade Metz, "How Facebook Moved 20 Billion Instagram Photos Without You Noticing," Wired, June 26, 2014, https://www.wired.com/2014/06/facebook-instagram/ ("First, they moved [Instagram] into Amazon's Virtual Private Cloud"). It is important to note that despite the name, Virtual Private Cloud is operated and managed on AWS' public cloud. It should not be confused with the term "private cloud" introduced earlier.

VPC offered more control for migrations out of Amazon's cloud. To facilitate the move from Amazon's cloud to the VPC, the engineers had to build their own custom networking tool, which they called Neti.[116] From there, the Facebook team migrated Instagram from VPC to Facebook's data centers.

81. This example demonstrates that high-growth, consumer-facing platforms with significant user data and real-time functionality can be migrated across cloud and organizational environments. Similarly, migrating AdX and DFP to a new owner would involve porting backend services,[117] but this example demonstrates that it is achievable even under tight performance and availability requirements. Furthermore, improvements in modern cloud migration technology would lead to fewer issues than Facebook encountered here.[118]

2) <u>Facebook Acquisition of WhatsApp</u>

82. Facebook acquired the instant messaging app WhatsApp in 2014. Facebook kept WhatsApp servers on IBM's cloud, the environment on which WhatsApp ran before the acquisition, during the software migration of Instagram.[119] Between 2016 and 2017, Facebook began migrating the WhatsApp infrastructure from IBM's cloud to Facebook's own data centers in a move similar to Instagram. While the exact timeline was never publicly released, the migration was complete by 2019. During this transfer, Facebook moved data and traffic from 1.5 billion users across data centers without disruption in service.[120] Facebook began by replicating data

---

[116] Instagram Engineering, "Migrating from AWS to FB," Medium, June 25, 2014, https://instagram-engineering.com/migrating-from-aws-to-fb-86b16f6766e2 ("So we developed Neti — a dynamic iptables manipulation daemon, written in Python, and backed by ZooKeeper.").

[117] *See* Section IX.C.5.

[118] *See, e.g.*, Cade Metz, "How Facebook Moved 20 Billion Instagram Photos Without You Noticing," Wired, June 26, 2014, https://www.wired.com/2014/06/facebook-instagram/ ("VPC didn't exist when Instagram was founded in 2010. Today, if other startups build on VPC from the beginning, they can avoid the extra[] steps that complicated Instagram's migration.").

[119] Jordan Novet, "Facebook is planning to move WhatsApp off IBM's public cloud, source says," CNBC, last modified June 7, 2017, https://www.cnbc.com/2017/06/07/facebook-planning-to-move-whatsapp-off-ibms-public-cloud.html ("Other companies that Facebook has acquired over the years initially used similar IBM SoftLayer servers, and Facebook was quick to bring them in-house, another source said. But with WhatsApp, Facebook has taken a much longer time. Part of the reason was prioritization. When Facebook bought WhatsApp, it was already in the midst of a yearlong migration of its Instagram photo-sharing app, which it acquired in 2012, from AWS to its own data centers.").

[120] Igor Istocniks, "How WhatsApp moved 1.5B users across data centers" at Code Beam SF, Code Sync, May 21, 2019, https://codesync.global/media/how-whatsapp-oved-1-billion-users-across-data-centers/.

being sent to the WhatsApp servers and directing it to Facebook's own servers.[121] Once it was clear that the transfer was successful, Facebook was able to cut the replication traffic and turn off WhatsApp data centers. This migration required translating systems between two different programming languages.[122] The migration also required successfully maintaining end-to-end encryption throughout the process.[123]

83. Facebook's migration of WhatsApp's deployment shows that even globally scaled, encrypted communications services can be re-platformed securely and without downtime. This example reinforces the feasibility of separating AdX and DFP, which also serve large-scale, latency-sensitive advertising workloads with a high need for reliability.

3) eBay's Sale of StubHub

84. eBay's sale of StubHub to Viagogo is another example of a large-scale software migration.[124] StubHub is a ticket exchange and resale company that was acquired by eBay in 2007 before being sold to Viagogo in 2020. This sale has technical similarities to Plaintiffs' proposed remedies.

85. StubHub received a copy of eBay's auction mechanism that StubHub adapted to its systems. In the interim, eBay retained the auction mechanism.[125] The migration of StubHub also

[121] Igor Istocniks, "How WhatsApp moved 1.5B users across data centers" at Code Beam SF, Code Sync, May 21, 2019, https://youtu.be/93MA0VUWP9w?si=Q77mvNiSEbAnTrgh&t=287.
[122] Erlang, which was used by WhatsApp, needed to communicate with Facebook's preferred language, C++. *See* Igor Istocniks, "How WhatsApp moved 1.5B users across data centers" at Code Beam SF, Code Sync, May 21, 2019, https://youtu.be/93MA0VUWP9w?si=y3Y8kRkkAJnjxVeb&t=391.
[123] Natasha Lomas, "WhatsApp completes end-to-end encryption rollout," TechCrunch, April 5, 2016, https://techcrunch.com/2016/04/05/whatsapp-completes-end-to-end-encryption-rollout/ ("It's a security project that's taken around a year and a half to complete, but messaging giant WhatsApp has now fully implemented strong end-to-end encryption on its platform and across all mobile platforms for which it offers apps.").
[124] eBay, "eBay Completes Sale of StubHub," eBay Press Release, February 13, 2020, https://www.ebayinc.com/stories/news/ebay-completes-sale-of-stubhub/.
[125] Competition & Markets Authority, "COMPLETED ACQUISITION BY PUG LLC (viagogo) OF THE STUBHUB BUSINESS OF EBAY INC.," https://assets.publishing.service.gov.uk/media/60702d63d3bf7f400a6b2feb/Final_undertakings_.pdf ("The Parties shall complete the migration of the StubHub Platform to the Google Cloud Platform. Once the abovementioned migration has been completed, the Parties shall cause the Duplicate Platform to be created."); *see also* the definition of "Duplicate Platform" ("The Duplicate Platform will contain the following data: (i) all Active Listings as at the Completion Date where the event location is within North America; (ii) all User Data of NorAm Buyers; (iii) all User Data of NorAm Sellers; (iv) the Transaction Data and Associated Listings in respect of transactions concluded between NorAm Buyers and NorAm Sellers; (v) the Unassociated Listings of NorAm Sellers; (vi) the Analytics Data, other than the International Analytics Data; (vii) the NorAm Catalogue Data; and (viii) the Platform Data.").

allow publishers to access data generated in DFP or AdX from their inventory in the same format as Google can access said data."[183]

133. These proposed data sharing remedies can be satisfied by building Application Programming Interfaces (APIs) into Google's existing software. In this section, I will discuss the timeline and milestones that would be needed to develop APIs to satisfy these components of the proposed remedies.

A. Google Has Existing APIs for Data Sharing That Can Be Adapted for Deployment

134. An API is a commonly used software component that can be used to send instructions or data from one program to another in an efficient manner. To build an API, an engineer distills information from a software program into a standardized format, so that any other program accessing the data will know exactly how it will receive the instructions or data. APIs can be used by an organization internally or can be made accessible to outside organizations and individuals. APIs are regularly developed by software companies, and it is less costly for an acquirer to access an API than to migrate the codebase behind the API. This is because accessing an API would not require any transferring of code to a new environment, and any API development would occur within the original organization's own environment. API development is a very common engineering task that many software engineers would be able to do.

135. Google offers many public-facing APIs and is very experienced in creating them for a variety of needs.[184] Both Section II, Phase 1 and Section V of the Plaintiffs' Revised Notice of Proposed Remedies ask Google to share data with outside organizations and publishers. An API would be well-suited for this kind of task and Google can readily build the APIs to satisfy the remedies for a few reasons.

---

[183] *See* Plaintiffs' Revised Notice of Proposed Remedies, p. 14, ECF No. 1482 ("Google should be required to allow publishers to access data generated in DFP or AdX from their inventory in the same format as Google can access said data.").

[184] "Google APIs Explorer," Google APIs Explorer, accessed June 24, 2025, https://developers.google.com/apis-explorer (describing 295 different Google APIs).

136. First, Google has already developed APIs to share advertising data with external organizations. As part of Google's 2021 settlement with the French Competition Authority, Google agreed to give "equal access to data related to outcomes from [Ad Manager auctions]."[185] The resulting Ad Manager's Data Transfer reports provide detailed data to advertisers that is updated every hour and includes information such as impressions, clicks, events, and user data.[186] While the type of data shared with publishers and the industry organizations administering the final publisher ad server auction would be different, the development of the respective APIs would essentially just require modifying the provided data fields, which is a trivial engineering task.

137. Second, Google would be sharing data that Google already uses internally. Google's own advertising platforms use APIs to communicate and share data with each other.[187] Section V.4 of the Plaintiffs' Revised Notice of Proposed Remedies specifically requests that the data in that remedy be shared "in the same format as Google can access said data." This means that Google would only need to adapt its existing APIs to be publicly accessible. [redacted].[188] Likewise, Section V.1 of the Plaintiffs' Revised Notice of Proposed Remedies requests data used to improve auction logic, so Google would be able to share the same data it currently uses for auction logic.[189]

---

[185] Google, "Commitments offered by Google under Article L.464-2, III of the French Commercial Code," p. 3, February 15, 2021.

[186] *See* "Ad Manager Data Transfer reports," Google Ad Manager Help, accessed June 24, 2025, https://support.google.com/admanager/answer/1733124?hl=en#zippy=%2Cabout-the-data-contained-in-data-transfer-files%2Cdownload-a-sample-file%2Chow-files-are-delivered%2Cfile-names%2Cdata-transfer-files-in-the-ad-request-process%2Cstore-files-locally%2Clearn-about-the-bigquery-data-transfer-service%2Cmake-large-data-transfer-files-easier-to-process.

[187] *See, e.g.*, GOOG-AT-MDL-B-009832326, at -391 (05/04/2023) ([redacted]).

[188] GOOG-AT-MDL-B-009828561, at -561–63 ("[redacted]").

[189] *See* GOOG-AT-MDL-000987584, at -584 (08/15/2017) ("We currently have two DataTransfer products, one for impression level AdX data (AdX DT) and one for impression-level DFP data (DFP DT)….We will be adding a new Bid-level DFP DataTransfer pipeline alongside the existing Backfill and Reservation pipelines.").

138. Lastly, Google will be deploying these APIs in its own environment. This simplifies the development process because Google's engineers will be working with source code and in an environment that they are experienced in.

B. [redacted]

139. [redacted].[190] [redacted]:[191]

- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]

140. [redacted].[192] [redacted], I find 18 months to be a reasonable estimate for the maximum amount of time required for the API development of a Prebid integration.[193] Importantly, the development of this API can be done in parallel with the open-source auction, as reflected in Figure III.1. I note that GOOG-AT-MDL-B-009832487 [redacted]

[190] GOOG-AT-MDL-B-009832482, at -483–86 ("[redacted]").

[191] GOOG-AT-MDL-B-009832487, at -513 (10/11/2024).

[192] Although in my experience, completing the engineering design of an API of this nature should take closer to 3 months rather than the [redacted]

[193] GOOG-AT-MDL-B-009832487, at -513 (10/11/2024) ("[redacted]").

which is longer than my estimate. However, I find that the [redacted] is more in line with what I would expect based on my experience.

VIII. THE OPEN SOURCE FINAL AUCTION LOGIC CAN BE DEPLOYED BY A THIRD-PARTY AT SCALE WITHIN TWO YEARS

141. Plaintiffs' proposed second phase of the DFP divestiture requires Google to separate out the "Final Auction logic" from DFP, provide it to third parties under an open source license, and provide an API that would allow a third party industry organization or other third party chosen by the publisher to administer the final auction outside of Google's control.[194] This third-party organization would be responsible for fulfilling the role of Google in providing this auction service. In this section, [redacted]

A. Google Overestimates the Time Necessary to Deploy the Open Source Final Auction Logic at Scale by a Third-Party

142. [redacted] I believe that this is an overestimate, and I expect this work can be completed in two years, as discussed below.

143. The open-sourcing process begins with the identification of the auction logic, through a process similar to what is described in Section IX.B. As I discuss in Section IX.B.2, this process is part of ongoing service monitoring, and even if no prior work has been performed, it would take at most one month.

144. The open-sourcing process typically involves establishing a governance body to manage the source code.[196] Establishing a governing body is a process that can be performed concurrently with the software split and migration and thus would not extend the timeline for

[194] Plaintiffs' Revised Notice of Proposed Remedies, pp. 10–11, ECF No. 1482.
[195] GOOG-AT-MDL-B-009832487, at -514 (10/11/2024) ("[redacted]").
[196] Nick Vidal, "What Is Open Governance? Drafting a charter for an Open Source project," Open Source Initiative, May 9, 2023, https://opensource.org/blog/what-is-open-governance-drafting-a-charter-for-an-open-source-project ("Building a healthy Open Source community is much more than just choosing an Open Source license for the project. It involves creating a contributing guide, adopting a code of conduct, and establishing an open governance structure that allows all members to actively participate in and contribute to the project.").

the software migration. A technical monitor can also provide input when selecting a governing body.

145. Once Google has open-sourced the Final Auction Logic, there are several milestones that a third-party organization would need to achieve in the process of successfully deploying and administering it. In this section, I walk through these milestones and an estimated timeframe for each step.

1) Timeline and Milestones

146. [redacted]

[redacted][197]

[redacted]

147. [redacted]

[197] GOOG-AT-MDL-B-009832487, at -501 (10/15/2024).

[redacted][198]

[redacted]

148. There are several milestones that an organization would need to achieve to successfully deploy open-source software. The Linux Foundation is one of the leading open-source projects and has established a standardized process for open sourcing software. In Table VIII.1, I map these processes to [redacted]. Note that this table specifically focuses on steps that a third-party would need to do to deploy the software assuming the source code has already been open sourced by the previous owner. It does not include milestones that the Linux Foundation includes for the previous owner.

[198] GOOG-AT-MDL-B-009832487, at -504 (10/11/2024).

Table VIII.1: Processes Involved to Open-Source Software and [redacted][199] [redacted]

Milestone	Purpose and Description	[redacted][200]
Infrastructure	The third party would need to identify the technical requirements of the Final Auction Logic and acquire the infrastructure necessary (such as cloud compute) to host the final auction logic.	[redacted]
Legal Analysis	The third party would need to perform a legal review of the licensing of the open-source code to ensure that any deployments would satisfy the legal requirements of the license(s).	
Code Modifications	The third party would need to modify the source code to be able to perform the necessary tasks.	
Testing	Once implemented, the third party would go through a process of debugging and run tests including unit tests and shadow tests to ensure the robustness of the deployment.	
Launch and Maintenance	The open-source project's governing body would continue to manage the source code. The third party would need to stay up to date with any updates and have a process to test and incorporate these updates.	

149. [redacted][201] [redacted]

[199] *See* Christine Abernathy et al., "Starting an Open Source Project," The Linux Foundation, accessed July 6, 2025, https://www.linuxfoundation.org/resources/open-source-guides/starting-an-open-source-project.
[200] GOOG-AT-MDL-B-009832487, at -501–04 (10/11/2024); *See* Figure VIII.3.1.
[201] GOOG-AT-MDL-B-009832487, at -501 (10/11/2024).

[redacted][202]

150. [redacted]

151. [redacted]

152. During the deployment of the Final Auction Logic, the governing body would be able to work towards many of these milestones simultaneously. For example, the processes of building out infrastructure to host the auction, performing legal review, and modifying code can be done in parallel. Additionally, the processes of testing and launch and maintenance would be similar to the process of deploying a software product developed in-house, and they are accounted for in this timeline. [redacted][203] I discuss this feasibility in the next section.

153. I visualize these proposed efficiencies in Figure VIII.3. This timeline demonstrates that through certain efficiencies, including community engagement and deployment to a third-party host, Google can open source the Final Auction Logic in two years.

[202] GOOG-AT-MDL-B-009832487, at -503 (10/11/2024).

[203] *See* GOOG-AT-MDL-B-009832487, at -501 (10/11/2024) ("[redacted]").

Figure VIII.3: [redacted][204]

[redacted]

IX. THE MIGRATION OF ADX AND DFP (AS STIPULATED BY PHASE 3 OF DFP DIVESTITURE) SHOULD EACH TAKE TWO YEARS OR LESS

154. In this section, I describe the process by which software application deployments of AdX and the remainder of DFP can be migrated to an acquirer's preferred environment, with an initial move from Google's private cloud to Google's public cloud, GCP.[205] I also provide estimates of the time and resources needed to successfully execute these migrations. I estimate a timeline of 15-18 months is necessary for the migration of AdX supported by at most 80 engineers, and a timeline of 19-24 months is necessary for DFP migration during Phase 3 of divestiture, supported by at most 80 engineers.

155. I note that Plaintiffs' proposed remedies call for an immediate AdX divestiture and DFP divestiture over three phases. This section discusses the work and timeline for migration of AdX and the remainder of DFP that will be in effect if Phase 3 of divestiture is ruled necessary

[204] Figure generated using values from GOOG-AT-MDL-B-009832487, at -504 (10/11/2024).

[205] *See* Plaintiffs' Revised Notice of Proposed Remedies, pp. 8–11, ECF No. 1482 (describing the proposed divestment process). In the case of Section II, I opine only on the *third phase* of full divestiture, as the *first phase* and *second phase* require the creation of new APIs, which does not require the migration of the DFP software application to an acquirer's environment. For a third-party to run an open source auction as described in Phase II.2 ("separate out the functionality that performs the final auction within the publisher ad server.... then be required to provide this auction-logic code under an open source license to industry organizations or participants"), that third-party would be required to deploy an application of the final auction logic. This would be a new deployment which may be similar to the stages I discuss here but would ultimately follow a different process.

by the court after the first two phases of DFP divestiture, discussed in Section VII and Section VIII. In the event that it is not deemed necessary to proceed to Phase 3 of DFP's divestiture, my estimates for the previous two phases of DFP's divestiture are not affected. AdX migration and Phases 1 and 2 of DFP divestiture can progress in parallel, and AdX migration may be completed prior to the start of DFP migration.

156. In developing my estimates for the migration of AdX and DFP, I consider (1) [redacted] (2) comparisons to software migrations in industry, (3) my review of code statistics from Google's AdTech systems, and (4) my own experience of large-scale code and data migrations and deployments.

A. [redacted]

157. [redacted][206] [redacted][207] [redacted]

158. [redacted][208] [redacted]

[206] *See* GOOG-AT-MDL-010415486, at -514 (02/28/2022) [redacted]").

[207] GOOG-AT-MDL-B-009832783, at -816 (04/10/2023) ("[redacted]").

[208] GOOG-AT-MDL-B-009832019, at -026 ("[redacted].").

[redacted].

159. [redacted][209] [redacted][210], [redacted][211]

160. [redacted][212] [redacted][213]

161. [redacted][214] [redacted]

[209] GOOG-AT-MDL-B-009832019, at -027.
[210] GOOG-AT-MDL-B-009832007, at -007.
[211] GOOG-AT-MDL-B-009832019, at -027.
[212] [redacted] *See* GOOG-AT-MDL-B-009832019, at -027.
[213] *See, e.g.*, GOOG-AT-MDL-B-009832019, at -027 ("[redacted]").
[214] *See* GOOG-AT-MDL-B-009832019, at -026.

[redacted]

162. [redacted].[215] [redacted].[216] [redacted][217] [redacted]

163. Although a common description applies to the process of migrating the software application deployments of AdX and DFP, the timeline and complexity of each migration is dependent on the deployment and environment of the product at Google and, importantly, on the environment of the acquirer of each product. [redacted][218]

[215] GOOG-AT-MDL-B-009828652, at -674 ([redacted]).

[216] GOOG-AT-MDL-B-009828652, at -674.

[217] GOOG-AT-MDL-B-009828652, at -675.

[218] GOOG-AT-MDL-B-009832487, at -505 (10/2024) ([redacted]).

164. As noted in Section III, my timeline scopes the time and headcount requirements for a migration to GCP first, as the number of shared internal services between Google's private and public cloud will support smooth migration.[219] In the future, an acquirer may choose to migrate the software to its preferred software environment. The timeline for this future migration may depend on the complexity of the software environment chosen by the acquirer and how different it is from GCP. Once the software has been successfully migrated to GCP, the process of migrating the software to another cloud service provider becomes significantly easier and is a standard practice among cloud service providers.[220]

165. I describe the timeline for migration in four stages: Deployment Analysis, Technical Decoupling, Testing and Debugging, and Final Deployment. The stages I describe are standard for software migrations. For example, Google describes the steps required for migration to be, Migration Preparation, Assess, Plan, Deploy, and Optimize.[221] In Table IX.1 below, I map my proposed stages to Google's suggested steps for a migration.

Table IX.1 – Proposed Stages for AdX and Phase 3 of DFP Divestiture and Google Suggested Migration Steps

Proposed Stages for Divestiture	Google Suggested Migration Steps[222]
Deployment Analysis	Migration tools architecture (Architecture of tools constituting a migration factory) Inventory (List of workloads that will be migrated to Google Cloud) Team Composition (Team composition and roster with contact details)

[219] *See* Section VI.A.
[220] Ramy Afifi, "Migrate compute from Google Cloud Platform (GCP) to AWS using AWS Application Migration Service," AWS, September 28, 2023, https://aws.amazon.com/blogs/storage/migrate-from-google-cloud-platform-gcp-to-aws-using-aws-application-migration-service/ ("Customers using Google Cloud Platform (GCP) might explore the option of spreading or transitioning their cloud usage away from GCP to alternative providers for various reasons, including cost evaluations, data centralization, or changes in business requirements.").
[221] "Execute your migration," Google Cloud, last updated July 2, 2025, https://cloud.google.com/migration-center/docs/migration-execution ("After you have completed the migration planning and preparation phase,…[1] Assess [2] Plan [3] Deploy [4] Optimize").
[222] "Execute your migration," Google Cloud, last updated July 2, 2025, https://cloud.google.com/migration-center/docs/migration-execution.

Technical Decoupling	Migration checklist (Checklist for use before and during the migration sprint) Sprint runbook (Execution guidelines for migrating each workload) Migration plan (Step-by-step migration process to be followed during the migration sprint) Network and security rules (List of all of the firewall rules for ingress and egress on Google Cloud, DNS changes during migration to Google Cloud) Testing and validation (Test plan to validate functional and non-functional requirements) Team composition (Team composition and roster with contact details)
Testing and Debugging	Migration tools architecture (Architecture of tools constituting a migration factory) Migration checklist (Checklist for use before and during the migration sprint) Migration plan (Step-by-step migration process to be followed during the migration sprint) Network and security rules (List of all of the firewall rules for ingress and egress on Google Cloud, DNS changes during migration to Google Cloud) Risks and mitigation (Possible risks during the migration sprint and mitigation steps) Testing and validation (Test plan to validate functional and non-functional requirements) Rollback plan (Rollback steps by workload) Team composition (Team composition and roster with contact details)

	Governance (RACI matrix of migration execution team, cadence and reporting, escalation resolution mechanisms)
Final Deployment	Migration tools architecture (Architecture of tools constituting a migration factory) Migration checklist (Checklist for use before and during the migration sprint) Migration plan (Step-by-step migration process to be followed during the migration sprint) Network and security rules (List of all of the firewall rules for ingress and egress on Google Cloud, DNS changes during migration to Google Cloud) Risks and mitigation (Possible risks during the migration sprint and mitigation steps) Testing and validation (Test plan to validate functional and non-functional requirements) Rollback plan (Rollback steps by workload) Team composition (Team composition and roster with contact details) Governance (RACI matrix of migration execution team, cadence and reporting, escalation resolution mechanisms)

166. Figure IX.1 and Figure IX.2 below summarize my estimated timelines for each of these stages. I detail these estimates in the following sections.

Figure IX.1: Estimated Timeline of AdX Migration

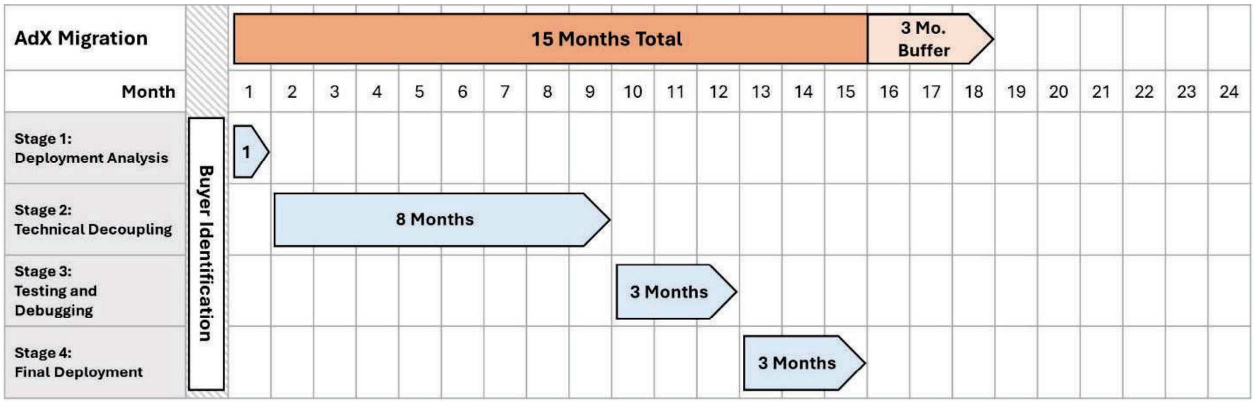
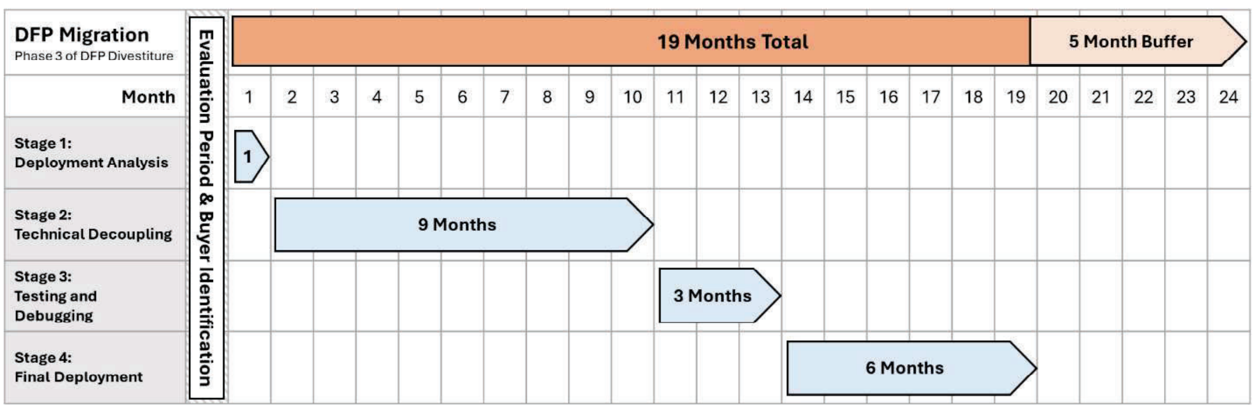


Figure IX.2: Estimated Timeline of DFP Migration (Phase 3)



B. Stage 1: Deployment Analysis

167. The first stage is Deployment Analysis. In Deployment Analysis, before software engineers make changes to the existing software, a small team of engineers will analyze Google's software and deployment systems to identify the relevant components to copy and migrate to replicate the necessary functionality of AdX and DFP.

168. During their preliminary analysis of the codebase, engineers can determine the nature of the "complexity factors," such as the configuration of servers used in deployment,[223] for that codebase. This will inform an updated estimate of the upfront costs associated with migrating

[223] I describe other example of complexity factors in Section IV.B.3.

the codebase. As demonstrated in Section VI.A, Google's current deployments of AdX and DFP indicate the ability to mitigate complexity and conduct migration to GCP swiftly. For AdX, this analysis can also begin in parallel with the identification of a buyer.

169. From my experience in industry, I understand that deployment analyses are a fundamental step in the software migration process. In fact, Google suggests customers perform deployment analyses before migrating their software to GCP.[224]

1) <u>Network analysis of application interactions</u>

170. Before beginning any process of technical decoupling, engineers can begin the analysis process for AdX and DFP by establishing a clear, data-driven understanding of how the system behaves under real-world production traffic.[225] The goal is to determine what "normal" looks like by capturing request patterns, current throughput, latency distribution, error rates, and resource usage, so that any later optimization or migration can be measured against an agreed-upon baseline.[226]

---

[224] "Migrate to Google Cloud: Assess and discover your workloads," Google Cloud, last modified August 2, 2024, https://cloud.google.com/architecture/migration-to-gcp-assessing-and-discovering-your-workloads ("Discovering your workloads and services inventory, and mapping their dependencies, can help you identify what you need to migrate and in what order. When planning and designing a migration to Google Cloud, you first need a deep knowledge of your current environment and of the workloads to migrate.").

[225] This is referred to as "Static Analysis." In Static Analysis, engineers first run the existing binaries in Google datacenters, gathering at least one to two weeks of metrics that cover both peak and off-peak periods. During this window they track queries per second, tail latencies relative to typical traffic, and error frequencies, while correlating anomalies with traffic surges or downstream-service issues. Simultaneously, they review CPU, memory, garbage-collection pauses, and thread-pool saturation on production instances via Borgmon/Monarch or Cloud Monitoring dashboards to spot capacity constraints.

[226] As part of establishing a performance baseline, the engineers will also need to perform a performance and latency analysis to identify current bottlenecks in the code and infrastructure, understand how the current system scales as load increases, and establish expected latency thresholds. Performance and latency analysis includes a set of key tasks: CPU profiling assesses the system's CPU usage under realistic loads and identifies functions or components of the system that use disproportionate CPU; memory and resource profiling helps engineers understand memory usage and detect inefficient memory allocations or errors; engineers use latency breakdowns to understand the time spent on each stage of a request; and load and scalability testing allows engineers to perform stress tests of the system that simulate typical traffic or heavy traffic and identify system capacity limits. Understanding the baseline state of the system within Google's environment will allow the engineers involved in the migration to specify the necessary status of the system when the migration is complete.

171. In addition to establishing system baselines, engineers involved in this process will need to strengthen the system's observability to allow engineers to continuously monitor system health, quickly diagnose issues, and rapidly assess impacts of any system changes. [227]

172. Finally, the team of engineers involved in the analysis stage of the migration will need to follow standard processes to identify the components relevant to the migration and understand all relevant dependencies. Dr. Weissman discusses the detailed analyses involved in this process;[228] I find that his analysis is sound, and I adopt it in evaluating the work to be performed in this stage.

2) Timeline and Resources for the Deployment Analysis Stage of AdX and DFP Migration

173. [redacted] This is because developing an understanding of how the system runs under real-world production traffic, an important part of deployment analyses process, is also part of ongoing service monitoring.[229] [redacted].[230] [redacted]

174. I provide my timeline and resource estimates assuming that no work has been completed, and therefore this can be considered a conservative estimate. Based on my experience with

[227] An observability system of this sort is likely to track the key metrics included in a performance baseline: latency, traffic, errors, and resource saturation. Good observability systems include well-structured logs that are easy for engineers to query as well as comprehensive dashboards that allow engineers and SREs to easily view system metrics. These sorts of systems should also include alerts that notify engineers when metrics are sufficiently different from baselines.

[228] Expert Report of Dr. Jon Weissman, July 7, 2025. Section VI.

[229] "Continuously monitor and improve performance," Google Cloud, last reviewed December 6, 2024, https://cloud.google.com/architecture/framework/performance-optimization/continuously-monitor-and-improve-performance ("After you deploy applications, continuously monitor their performance by using logs, tracing, metrics, and alerts.").

[230] GOOG-AT-MDL-B-009835902 (06/23/2025).

large-scale software migrations in industry, I find that deployment analysis typically takes at most one month and requires at most 12 software engineers.

C. Stage 2: Technical Decoupling

175. The next stage of AdX or DFP migration would involve technical decoupling of the software as well as modifying and implementing additional services to support the required software components in the new environment. Decoupling is the process of modifying a software component's source code such that it no longer interfaces with another software component.[231] Decoupling work can be done without interrupting the system that is being migrated.[232] This stage also involves modifying and implementing additional services to support the required software components in the new environment.

176. In this section, I discuss the five workstreams that will be performed during Technical Decoupling: (1) Setup Environment and Infrastructure, (2) Disconnect Communication to External Components, (3) Modify Internal Communications, (4) Replace Mocks and Stubs, and (5) Transfer Data to Acquirer's Environment. These workstreams are standard in software migrations, though I note that other similar steps can be taken to migrate software which require a similar amount of time. In Table IX.2 below, I map the proposed workstreams for technical decoupling to Google suggestions for migrating software to GCP. I note that the work described in workstreams (2)-(4) is standard in the industry and is commonly referred to as a "Strangler Fig pattern."[233]

<u>Table IX.2 – Proposed Workstreams for Technical Decoupling of AdX and DFP Migration and Google Suggested Migration Steps</u>

[231] "Google Cloud Well-Architected Framework," Google Cloud, last reviewed October 11, 2024, https://cloud.google.com/architecture/framework ("Decoupling is a technique that's used to separate your applications and service components into smaller components that can operate independently.").
[232] *See* Section V for examples of software migrations that did not cause any downtime for the product.
[233] *See* "Refactor a monolith into microservices," Google Cloud, last reviewed June 26, 2024, https://cloud.google.com/architecture/microservices-architecture-refactoring-monoliths; *see also* Samir Behara, "Monolith to Microservices With the Strangler Pattern," DZone, December 12, 2018, https://dzone.com/articles/monolith-to-microservices-with-the-strangler-patte.

Proposed Workstreams for Technical Decoupling	Google Suggested Migration Steps[234]
(1) Setup Environment and Infrastructure	Designing a Landing Zone in Google Cloud[235] Team Composition (Team composition and roster with contact details) Rollback plan (Rollback steps by workload)
(2) Disconnect Communication to External Components	Network and security rules[236] Testing and validation (Test plan to validate functional and non-functional requirements) Team composition (Team composition and roster with contact details)
(3) Modify Internal Communications	Part of proposed step-by-step migration plan that Google recommends developing before a migration[237] Testing and validation (Test plan to validate functional and non-functional requirements) Team composition (Team composition and roster with contact details)
(4) Replace Mocks and Stubs	Part of proposed step-by-step migration plan that Google recommends developing before a migration[238]

[234] "Execute your migration," Google Cloud, last updated July 2, 2025, https://cloud.google.com/migration-center/docs/migration-execution.

[235] "Landing zone design in Google Cloud," Google Cloud, last reviewed October 31, 2024, https://cloud.google.com/architecture/landing-zones ("A landing zone is often a prerequisite to deploying enterprise workloads in a cloud environment.").

[236] "Execute your migration," Google Cloud, last updated July 2, 2025, https://cloud.google.com/migration-center/docs/migration-execution ("Network and security rules[:] List of all of the firewall rules for ingress and egress on Google Cloud DNS changes during migration to Google Cloud").

[237] "Execute your migration," Google Cloud, last updated July 2, 2025, https://cloud.google.com/migration-center/docs/migration-execution ("Migration plan[:] Step-by-step migration plan (process) to be followed during the migration sprint").

[238] "Execute your migration," Google Cloud, last updated July 2, 2025, https://cloud.google.com/migration-center/docs/migration-execution ("Migration plan[:] Step-by-step migration plan (process) to be followed during the migration sprint").

	Network and security rules (List of all of the firewall rules for ingress and egress on Google Cloud, DNS changes during migration to Google Cloud) Risks and mitigation (Possible risks during the migration sprint and mitigation steps) Testing and validation (Test plan to validate functional and non-functional requirements) Team composition (Team composition and roster with contact details)
(5) Transfer Data to Acquirer's Environment	Part of proposed step-by-step migration plan that Google recommends developing before a migration[239] Migration plan (Step-by-step migration process to be followed during the migration sprint) Network and security rules (List of all of the firewall rules for ingress and egress on Google Cloud, DNS changes during migration to Google Cloud) Rollback plan (Rollback steps by workload) Testing and validation (Test plan to validate functional and non-functional requirements) Risks and mitigation (Possible risks during the migration sprint and mitigation steps)

177. This stage will require collaboration between software engineers at Google and the acquirer's company. Early technical collaboration with Google engineers and allowing the acquirer's engineers access to interface specifications is essential to draw clean system boundaries and avoid late-stage surprises.

[239] "Execute your migration," Google Cloud, last updated July 2, 2025, https://cloud.google.com/migration-center/docs/migration-execution ("During the deployment phase, your migration team executes the migration plan and removes any critical issues."); *see also* "Execute your migration," Google Cloud, last updated July 2 2025, https://cloud.google.com/migration-center/docs/migration-execution ("Sprint runbook[:] Execution guidelines for migrating each workload…Migration plan[:] Step-by-step migration plan (process) to be followed during the migration sprint").

1) Timeline and Resources for Technical Decoupling Stage of AdX Migration

178. Based on my experience with large-scale software migrations in industry, I find that technical decoupling of this size typically takes around eight months and should not require more than 80 software engineers.

2) Timeline and Resources for Technical Decoupling Stage of DFP Migration

179. Based on my experience with large-scale software migrations in industry, I find that technical decoupling of this size typically takes around eight months and requires about 80 software engineers. [redacted]. With this in mind, I include an additional buffer of one month to account for this higher complexity.

180. This stage will consist of five workstreams. Each workstream, a brief description of its goals, and potential factors that will influence the complexity of the workstream, are listed below. I also describe the milestone reached at the conclusion of each workstream.

3) Workstream 1: Setup Environment and Infrastructure

181. The first engineering priority is to create a secure, production-grade "landing zone" where the separated binaries can run long before any code is ported. This involves choosing the target cloud (in this case, GCP as a first step) or on-premises substrate, preparing infrastructure-as-code templates, and mapping Google-specific middleware[240] to acquirer equivalents,[241] object storage, and a Prometheus and Grafana observability stack.[242]

182. Standing up the environment is accomplished by concrete engineering tasks.

---

[240] Google-specific middleware includes, but is not limited to, tools like Borg, Stubby, Spanner, Colossus, and Monarch.

[241] Equivalents to Google-specific middleware include Kubernetes, gRPC, CockroachDB, and CosmosDB.

[242] A skeletal CI/CD pipeline—with artefact signing and a small canary lane—should be online by week two, capacity should be pre-provisioned to about seventy percent of projected peak load, and golden-signal dashboards plus distributed traces need to be in place so that later migrations can be measured against a stable baseline.

183. Core hosting resources[243] can be built out in a sandbox that is isolated yet production-like. These environments are often called "canary" or "staging" environments.[244]

- Identity-and-access management is configured for both operations and development staff, mirroring Google's role hierarchy wherever practical to minimize porting effort.
- Operating-system images, shared libraries and service-orchestration policies can be cloned as closely as possible to Google's defaults so the binaries behave predictably. As discussed in Section VI.A., Google's internal OS is compatible with migrations to closely related OS distributions (e.g., from Linux).

184. Deployment regions can be chosen to match the existing deployment topology so that bidder traffic sees minimal additional network distance. Throughout, engineers track latent complexity that may carry hidden ties to Google infrastructure.[245]

185. As described previously in Section VI.A., migrating Google's systems to Google Cloud Platform would make the process meaningfully easier. If requested, an acquirer could also request GCP as an intermediate home for the system while it works to migrate the software to another system such as a competing public cloud platform. In this case, after the migration to Google Cloud Platform, the acquirer could migrate the software at its own pace and with its own goals while the product is fully operational. The assumption that Google's systems would be migrated to GCP will impact my estimated timelines for all downstream steps.

---

243 Such as Virtual Machines (VMs), container nodes, and service meshes.

244 Alec Warner and Štěpán Davidovič, "Canarying Releases," in *Google SRE Workbook*, ed. Betsy Beyer et al. (Sebastopol, CA: O'Reilly, 2018), https://sre.google/workbook/canarying-releases/ ("[C]anarying is a partial and time-limited deployment of a change in a service and its evaluation. This evaluation helps us decide whether or not to proceed with the rollout. The part of the service that receives the change is 'the canary,' and the remainder of the service is 'the control.' The logic underpinning this approach is that usually the canary deployment is performed on a much smaller subset of production, or affects a much smaller subset of the user base than the control portion. Canarying is effectively an A/B testing process.") (emphasis omitted). Google seems to use this terminology in its own deployments and API migrations. *See, e.g.*, GOOG-DOJ-AT-02447913 (use in deployments); GOOG-AT-MDL-004054765 (use in API migrations).

245 Latent complexity can often be found in inter-service dependencies, back-pressure rules, synchronous versus asynchronous paths and any long-tail cron or batch jobs.

190. There are two risks to monitor during this period. First, there may continue to be hidden dependencies on Google services.[250] There may also continue to be references to Google IP addresses that are hard-coded throughout the system that will need to be monitored.[251] Second, the performance of the overall system may appear degraded. It may be the case that until replacement services reach full scale (see Section IX.C.6), critical paths may experience added latency or reduced data freshness. As discussed in Section IV.B.3, careful phased rollouts with robust monitoring are typically used to manage these issues.

Milestone: All connections between the migrated product and non-transferred Google systems are fully severed or stubbed; verified by integration tests and compliance reviews confirming no unauthorized data flows or API calls remain.

5) Workstream 3: Modify Internal Communications

191. Building on the findings from stage one, this workstream evaluates all systems, services, and tools used by the product for its internal analysis and communication and determines if each interaction can stay as-is, must be rewritten, or demands a full replacement. The guiding principle is that *all* messages, RPCs, and background jobs must continue to function independent of Google's internal services.[252] This principle ensures that everything essential can continue to run once the internal communications between system components have been modified. As a result of Workstream 2, engineers will have a structured assessment that includes a catalog of every internal API endpoint, message queue, gRPC method, cron task, and shared-memory bus (all of these are ways in which components exchange information or coordinate work) that the product currently relies on for each item that is recorded from the dependency chain and grades the difficulty of decoupling.[253]

---

[250] Hidden dependencies on Google's services can be mitigated by continuous packet monitoring, coupled with dependency-scanner-based static analysis.
[251] For instance, contractual allow-lists hard-coded in bidder or partner configurations may still reference Google IP ranges. The business-development team coordinates updates with partners early, and a temporary dual-DNS scheme can translate legacy hostnames to new addresses during the cut-over window.
[252] Google internal services include, but are not limited to, Borg schedulers, Stubby headers, Google Pub/Sub topics, Monarch telemetry, and "Zeus" Zookeeper.
[253] Items from the dependency chain can be broken down into questions such as, does it transit Google Pub/Sub? Does it assume Borg-style service discovery? And does it authenticate with Google-internal metadata servers?

192. Where a service to be replaced is proprietary, a small team can conduct a planning exercise to select a substitute and documents its pros, cons, and migration steps.[254] Examples of services that will need to be replaced are logging, tracing, and audit services.[255] These can be supported through standard industry tools.[256] A decision about how to replace each service can be captured in a decision register so work on each service can be undertaken parallel to groups with minimal overlap or interference.[257]

Milestone: Internal systems and services can communicate seamlessly without relying on Google's proprietary frameworks or infrastructure, verified through a full-system dry run in the new environment under expected production conditions.

6) <u>Workstream 4: Replace Mocks and Stubs</u>

193. Once the internal connections between systems are updated, the temporary placeholders used for testing are replaced with the real, production-ready services.[258] The system is carefully switched over from a testing mode ("mock") to a live mode ("pod") in gradual, controlled steps to make sure everything works correctly. To check this, engineers replay past ad auction traffic to confirm the system behaves the same way it did before. Engineers also preload important data so that the first real user requests after the switch don't cause delays. Most of the complexity in this stage comes from differences in how the old and new systems store, retrieve, and process data.

194. The actual mechanics for this process are straightforward. Every new service is launched with three settings. First, it is *disabled*, which means the system is still using a test version. Second, it moves to *shadow mode*, where real traffic is sent to the new service behind the scenes, but its answers aren't actually used. This is to monitor performance of the service as it

[254] Stubby RPCs, for example, map cleanly to open source gRPC carried through an Envoy or AWS App Mesh data plane; "Zeus" (Google's internal Zookeeper fork) can be replaced by upstream Apache Zookeeper, Consul, or AWS Cloud Map; and Monarch counters move to Prometheus plus Grafana.
[255] These events are traditionally pushed into Google's Dapper/Monarch stack.
[256] For example, OpenTelemetry for monitoring, Loki or Elasticsearch for log aggregation and search, Prometheus for metrics collection, and Grafana for visualization and alerting.
[257] After protocols are chosen, engineers adjust service discovery and monitoring. All deployment descriptors that once pointed at Borg's DNS are rewritten to the DNS of the acquirer or Consul addresses. Sidecars inject mTLS certificates issued by the acquirer's PKI rather than Google's. Alerting rules in Prometheus mirror the old Monarch SLO dashboards so on-call staff will see familiar graphs the day traffic flips.
[258] These include, but are not limited to, real-time profile stores, budget ledgers, and creative fetchers.

would perform in a live system. Third, it becomes *primary*, meaning the new service is officially in use in the production system. The switch to primary happens gradually: first to 1% of traffic, then 5%, 20%, 50%, and eventually 100%.[259] To make sure everything is working as expected, the team can also run a set value (e.g., at least one million) of past ad auctions through the new system to test it thoroughly before going live.[260]

195. To prevent delays in live production, each new service must produce the same or better key performance metrics as the system it replaces, include structured logs with trace identifiers for debugging, and integrate with existing alert systems to ensure that on-call engineering teams experience a smooth and transparent handoff. For instance, each data store is proactively prepared by preloading the most frequently accessed data and is only then progressively loaded with the rest of the data in the background before the system officially goes live. In addition, safeguards such as load-shedding, circuit breakers, and automatic five-second fallback responses should be built in from the start. These measures ensure that if any one part of the system slows down, it does not disrupt the strict timing requirements of Google's global ad auction process.

196. One of the most technically complex aspects of this system migration involves reconciling differences in configurations and capabilities between the source and target systems. An acquiring company's infrastructure may be designed with different goals and priorities in mind, which are not inherently inferior, but represent different engineering trade-offs and design decisions. To bridge this divergence, engineers evaluate the requirements of the application and how they map to the behavior of the target system of the acquirer. Some adjustment of configuration parameters or possibly rewriting parts of the application code could be required in this work.

[259] This promotion is subject to an automated rollback if p99 latency regresses by more than five milliseconds, error rate exceeds 0.1 percent, or business KPIs (win-rate, revenue per mille) drift beyond a pre-agreed tolerance band. "p99 latency" refers to a common performance metric that measures how fast 99% of requests are completed.
[260] It is my suggestion that the decision deltas must remain below 0.01 percent and latency uplift below two milliseconds before the first canary gate opens.

Milestone: All placeholder components and temporary stubs introduced during decoupling are replaced with fully functional, production-grade services in the acquirer's stack, with passed acceptance tests and performance benchmarks.

7) Workstream 5: Transfer Data to Acquirer's Environment

197. In Workstream 5, key data is transferred from Google to an acquirer. This may include historical logs and live system state information such as auction events, bidder seats, user segments, and pacing budgets. The process ensures that data is transferred securely and systematically.

198. The transfer process follows a structured pipeline. Data are exported from Google's internal systems (e.g., Bigtable or Spanner) into neutral formats (e.g., Parquet or Avro), encrypted, and delivered to an acquirer. Each batch includes validation artifacts like checksums, record counts, and signed manifests.

199. Live data are transitioned gradually. For example, actively logged data that are key to AdX or DFP functionality, such as the streaming state of the services, are written to both Google's environment and a newly deployed acquirer's environment until the system is verified to receive all necessary data, at which point Google will no longer receive the data.

200. Several technical challenges must be managed. These include high log volumes, ongoing GDPR erasure requests,[261] encryption key transitions, and differences in identity and access control systems. Large data volumes may require special handling. If the network cannot handle the full load, teams may use hardware appliances for bulk transfer or break the job into region-specific export and import phases.[262]

[261] The General Data Protection Regulation, or GDPR, is a regulatory act in the European Union that, among other things, includes the Right to Be Forgotten. That is, an individual may request their personal data be deleted from a database. *See* "Regulation (EU) 2016/679," European Parliament, Art. 17, April 27, 2016, https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32016R0679.

[262] For example, Google Transfer Appliance or AWS Snowball; For streaming states held in MillWheel, the system enters a dual-write phase wherein every event is written simultaneously to Google storage and the new pipeline until watermark lag reaches zero seconds, at which point the Google writers are drained and decommissioned. A similar shadow-mode pattern applies to near-real-time click logs and impression beacons; traffic mirrors flow for twenty-four hours and reconciliation queries must show a byte-for-byte match before the switchover.

201. Migration complexity of data transfer depends heavily on an acquirer's environment. Key factors include whether the target stack runs on a public cloud, a private datacenter, or a multi-tenant SaaS platform. Each scenario introduces different technical constraints.

202. The operational demands of the system also increase migration difficulty. High daily data ingest rates[263] combined with live data erasure requests, require precise and repeatable deletion workflows. Additionally, data security and access models must be carefully aligned. Encryption key hand-offs between key management systems (KMS) and mismatches in identity and access management (IAM) models may require role mapping or token translation to ensure compliance.

203. Dry-run transfers mitigate these risks. One may first migrate a ten percent sample of all data, test every downstream analytics job, and transfer the full data set only after dashboards agree on record counts, schema structure, and business KPIs.

Milestone: All relevant product and operational data has been successfully migrated to the acquirer's environment, validated for completeness, schema integrity, and compliance with court-mandated privacy standards.

D. Stage 3: Testing and Debugging

204. After technical decoupling is complete, the third stage of migration would include testing all the components of AdX or DFP after they have been implemented in the acquirer's development environment. Stage 3 is the risk-burn-down phase: every component that now runs in the acquirer's environment is validated through unit, integration, load, stress, and A/B tests.[264]

205. Factors that influence the complexity of testing include the number of features and services that need to be tested, and the quality of the monitoring tools that were built in Workstream 1.

206. In my review of other large-scale software migrations in industry, I find Spotify's migration strategy from its own private cloud to GCP to be a reliable comparison for the effort that is

[263] "Ingest rate" is the speed and volume at which new data flows into the system.
[264] Traffic migrates in concentric rings: 0% (shadow), 1%, 5%, 20%, 50%, 100%, with automated rollback on any Service Level Objective or "SLO" breach.

required for testing and debugging in a software migration. In Spotify's case, it dedicated roughly 30% of its time during its two-week sprints to perform reliability testing.[265] However, in the case of an AdX or DFP migration from Google's private cloud to GCP, the two clouds share many components such as storage systems,[266] and thus directly using Spotify's percentage of time dedicated to testing would be an overestimate.

1) Timeline and Resources for Testing and Debugging Stage of AdX Migration

207. Based on my experience with large-scale software migrations in industry, I find that testing and debugging typically takes no more than three months and requires no more than 40 software engineers.

208. This timeline is roughly 18% of the total migration period, which, using Spotify's 30% benchmark and taking into account the shared components described in Section VI.A., indicates that my estimate is in a sensible range.

209. Further, this engineering estimate is roughly half of the number of engineers I estimated were required in the technical decoupling stage, which is in line with my experience performing large-scale software migrations in industry.

2) Timeline and Resources for Testing and Debugging Stage of DFP Migration

210. Based on my experience with large-scale software migrations in industry, I find that testing and debugging typically takes no more than three months and requires no more than 40 software engineers. [redacted] With this in mind, I include an additional buffer of one month to account for this higher complexity.

---

[265] *See* Google Cloud Tech, "Spotify's Journey to the Cloud (Cloud Next '18)," YouTube Video, July 26, 2018, https://youtu.be/5aBORQim-KM?si=NstturGYB0xg9Zqq&t=1194 ("Service Migration: Sprint work…Migration work: 50%… Reliability Testing: 30%… Education: 20%").
[266] As I discuss in Section VI.A.

APPENDIX B: CURRICULUM VITAE

Goranka Bjedov
1180 Saint Anthony Court
Los Altos, CA 94024
goranka@gmail.com

Education

Purdue University, West Lafayette, IN 47907
Master's of Science, Computer Science, May 1998

Clarkson University, Potsdam, NY 13676
Ph.D., Engineering Science - May 1992
Master's of Science, Civil Engineering - May 1988

Zagreb University, Zagreb, Croatia
Bachelor of Science, Civil Engineering - February 1986

Professional Experience

Facebook, Inc.
1 Hacker Way
Menlo Park, CA 94205

Performance and Capacity Engineer (August 2010- February 2019)
Technical lead of performance and capacity team:

- Analyzing and debugging performance problems.
- Modeling and predicting capacity needs of various services.
- Ensuring capacity needs (servers, network, power, data centers) are met.
- Supporting acquisitions and ensuring migrations to Facebook network
- Training all new engineers and managers in Capacity Engineering

Google, Inc.
1600 Amphitheatre Parkway
Mountain View, CA 94043

Senior Performance Test Engineer (January 2005 -August 2010)

Technical lead of performance team:

- Responsible for designing performance testing infrastructure relying on open source software and supporting all of Google's products and services.
- Performing experimental performance testing and producing system performance graphs for front end search properties, productivity tools and ads; diagnosing performance issues and working with respective development teams on fixing those; developing and scripting benchmarks used to catch or prevent performance degradations.

- Performance tests for new hardware versions of Google Search Appliance focused on memory and CPU performance.
- Performance consulting on internal products, work executed by respective development teams.

Related tasks:

- Performance and reliability analysis of back ends for all video products.
- Stabilizing and shortening release cycle for AdSense 3.0

Network Appliance
495 East Java Drive
Sunnyvale, CA 94089

Senior Engineer (May 2001 -November 2004)

Technical lead of Test Automation, Development and Analysis (TADA) Group:

- Responsible for designing, developing and executing a suite of reliability tests to predict product performance in the field, analyzing results and working with development teams on problem analysis and debugging.
- Proposing, creating and executing sanity test suites for file system development team to speed up release process - winner of 2004 Technical Excellence Award.
- Analyzing test procedures and practices and advising on major projects.

AT&T Labs - Internet Platform Technology Organization
2665 N. First St - Suite 300
San Jose, CA 95134

Technical Manager (May 1998 - May 2000)

Technical Lead and Manager for performance team:

- Primary responsibilities included all aspects of performance, security and API testing.
- Evaluation and selection of testing tools, design and implementation of testing process including test plans.
- Hiring and supervising engineers, creating progress reports and summaries of the testing process for the organization.
- Primary consultant to other AT&T Units requiring performance testing help and expertise.

Academic Experience

School of Civil Engineering and Department of Freshman Engineering
Purdue University
West Lafayette, IN 47907

Assistant Professor (September 1991 - May 1997)
Associate Professor (May 1997 - May 1998)

- Teaching C programming classes and computer tools classes.
- Advising graduate and undergraduate students; obtaining funding and conducting research;

publishing textbooks and articles; serving on committees.

Department of Civil and Environmental Engineering
Clarkson University
Potsdam, NY 13676

Instructor (August 1989 - September 1991)
Teaching Assistant (August 1986 -August 1989)

- Responsible for undergraduate programming class and sophomore engineering science classes as required by the department.
- Research in computational fluid mechanics, publishing journal articles and technical reports.

Publications

Author and co-author on two textbooks, numerous conference and journal papers and presentations and several invited talks and lectures at conferences and universities around the world. Numerous presentations related to performance engineering and capacity engineering, some of which are available on line.